Premium and Atrium using Unity Pro
AS-i Bus
User manual

October 2005 eng

35006196 03

a brand of Schneider Electric

Telemecanique

Table of Contents

Safety Information 9

About the Book 11

Part I The AS-i Bus 13

At a Glance 13

Chapter 1 General introduction to the AS-i bus 15

At a Glance 15
Reminder relating to AS-i and AS-i V2 buses 16
Overview of AS-i products from the Schneider catalog 17
Presentation of main constituent components 18
Example of AS-i bus topology 22
Main characteristics of the AS-i bus 23
Main Characteristics of the AS-i V2 Bus 25

Part II Hardware installation for AS-i power supply units and TSX SAY 100 and TSX SAY 1000 AS-i bus interface modules. 29

At a Glance 29

Chapter 2 AS-i bus power supply units 31

At a Glance 31
2.1 Introduction to AS-i Bus power supply units 32
At a Glance 32
General introduction to AS-i power supply units 33
Physical description of the TSX SUP A02 power supply module 34
Physical description of the TSX SUP A05 power supply block 35
Physical description of the support board 36
Catalog of AS-i power supply units 37
AS-i power supply: specific features 38
2.2 Installation of the AS-i bus power supply units 39
At a Glance 39
Dimensions/mounting of the AS-i TSX SUP A02 power supply unit 40

Dimensions/mounting of the TSX SUP A05 AS-i power supply unit 43
Summary of methods of fixing 44
2.3 AS-i power supply connections 45
At a Glance 45
TSX SUP A02 power supply connections 46
TSX SUP A05 power supply connections 48
General precautions. 52
2.4 Characteristics of AS-i bus power supply units 54
At a Glance 54
Electrical characteristics of AS-i power supply units TSX SUP A02/A05. 55
Physical characteristics of environment. 57

Chapter 3 AS-i bus interface module: TSX SAY 100 59
At a Glance 59
3.1 Description of the TSX SAY 100 Module. 60
At a Glance 60
Physical description 61
Mounting/installation 63
Connections. 64
Displaying module states. 67
Specific Display Panels on the TSX SAY 100 Module. 68
Technical characteristics 70
Safety of personnel 71
3.2 AS-i Bus diagnostics 72
At a Glance 72
Introduction to AS-i Bus diagnostics 73
Moving between the different display modes. 75
Displaying slaves on the AS-i bus 76
Displaying the state of the inputs/outputs of each slave 77
3.3 Operating modes of the TSX SAY 100 module 79
Operating modes of the TSX SAY 100 module 79
3.4 Precautions of use 81
24 V auxiliary supply 81

Chapter 4 AS-i V2 bus interface module: TSX SAY 1000. 83
At a Glance 83
4.1 Description of the TSX SAY 1000 Module. 84
At a Glance 84
Physical description 85
Assembly/installation 87
Connections. 88
Displaying Module States 91
Specific Display Panels on the TSX SAY 1000 Module. 92
Technical characteristics of As-i V2 bus 93
Personnel Safety 94
4.2 AS-i Bus Diagnostics 95

Introduction to AS-i Bus diagnostics 95
4.3 Operating Modes of the TSX SAY 1000 Module 97
Operating modes of the TSX SAY 1000 module 97
4.4 Precautions of Use 99
At a Glance 99
24 VDC Auxiliary Supply 100
Multiple Address Settings 101
4.5 AS-i V2 Certification 102
AS-i V2 Certification 102

Part III AS-i Bus software implementation 105
At a Glance 105

Chapter 5 Principles of the software implementation of the AS-i Bus . 107
At a glance 107
Presentation of the implementation of the AS-i bus 108
Architecture of TSX SAY 100 Module 110
Structure of an AS-i slave 112
Addressing of language objects associated with the slave devices connected to the AS-i bus. 114

Chapter 6 AS-i bus configuration 117
At a Glance 117
How to Declare an AS-i Communication Module in the PLC Rack 118
Description of the AS-i communication module configuration screen 119
How to Define a Slave Device on the AS-i Bus. 121
AS-i Bus Displayed in the Project Browser 124
How to modify the AS-i Bus software configuration 126
How to access the description of an AS-i slave 127
How to Define a New Slave Profile in a Private AS-i Catalog 129
How to modify AS-i slave general parameters: Automatic addressing 131
How to modify AS-i slave general parameters: Fallback mode. 132

Chapter 7 Debugging the AS-i bus 133
At a Glance 133
Introduction to the Debug function 134
Description of the debug screen of an AS-i module 135
How to access the functions of the module diagnostics and channel diagnostics of an AS-i device 137
Displaying the status of the slaves 139
How to Access Adjustment of an AS-i Device's Parameters. 140
How to Access the AS-i Channels' Forcing/Unforcing Function. 141
How to Access the SET and RESET Commands of the AS-i Channels 142
Automatic replacement of a faulty AS-i slave 143
How to insert a slave device into an existing AS- i configuration 144
How to modify the address of an AS-i device 145

Chapter 8 AS-i operating mode 147
At a glance 147
AS-i Operating Mode: General 148
AS-i protégé Mode 150
AS-i test de câblage mode 151
AS-i Offline Operating Mode 152
AS-i Data Exchange Off Operating Mode 153

Chapter 9 AS-i performance 155
Performances of the AS-i bus 155

Chapter 10 DFB for AS-i Security Monitor 157
AS-i security monitor DFBs 157

Chapter 11 Language objects of the AS-i bus 159
At a glance 159
11.1 Language objects and IODDT of the AS-i communication 160
At a glance 160
Presentation of language objects of the AS-i bus 161
Implicit exchange language objects associated with the application-specific function 162
Explicit exchange language objects associated with the application-specific function 163
Management of Exchanges and Reports with Explicit Objects 165
11.2 The language objects and generic IODDT applicable to all communication protocols 169
At a Glance 169
Details of IODDT implicit exchange objects of type T_COM_STS_GEN 170
Details of IODDT explicit exchange objects of type T_COM_STS_GEN 171
11.3 Language objects and IODDT's associated to the AS-i bus 173
At a glance 173
Details of the implicit exchange objects of T_COM_ASI_V1 type IODDTs 174
Details of the explicit exchange objects of the T_COM_ASI_V1 type IODDT . . 177
Details of implicit exchange objects of the AS-i function 179
Details of explicit exchange language objects for the AS-i function 182
Details of explicit exchange language objects for the AS-i function 184
Details of configuration objects for the AS-i function 186
11.4 The IODDT type T_GEN_MOD applicable to all modules 187
Details of the Language Objects of the IODDT of Type T_GEN_MOD 187

Part IV Implementation of AS-i V2 bus software 189
At a Glance 189

Chapter 12 Principles of the software implementation of the AS-i V2 Bus 191
At a glance 191

Presentation of the installation of the AS-i V2 bus 192
Architecture of the TSX SAY 1000 Module 194
Structure of a Standard Address AS-i 196
Structure of a Standard Address AS-i slave 198
Addressing of language objects associated with the slave devices connected to the AS-i bus. 200

Chapter 13 AS-i V2 bus configuration 203
At a Glance 203
How to declare an AS-i V2 communication module in the PLC Rack 204
Description of the configuration screen of an AS-i V2 communication module 205
How to define a slave device on the AS-i V2 bus 207
AS-i V2 Bus Displayed in the Project Browser 211
How to modify the AS-i V2 Bus software configuration 213
How to access the description of an AS-i V2 slave 214
How to define a new slave profile in the standard AS-i V2 catalog 216
How to modify the general parameters of an AS-i V2 slave: Automatic addressing 218
How to modify the parameters of an AS-i V2 slave 219
How to modify the parameters of an analog AS-i V2 slave 220
How to Modify the Parameters of an AS-i V2 Slave with Combined Parameters 221
AS-i V2 security device issues 222

Chapter 14 Debugging the AS-i V2 bus 223
At a Glance 223
Introduction to the Debug function 224
Description of an AS-i V2 module's debug screen 225
How to access module- and channel-diagnostics functions on an AS-i V2 device. 227
Displaying the slaves' status 229
How to access the adjustment of the parameters of an AS-i V2 device 231
How to access the AS-i V2 digital channels' forcing/unforcing function 233
How to access the SET and RESET commands of the AS-i V2 digital channels 234
How to modify the value of an analog channel 235
Automatic replacement of a faulty AS-i V2 slave 236
How to insert a slave device into an existing AS-i V2 configuration. 237
How to modify the address of an AS-i V2 device 238

Chapter 15 AS-i V2 operating mode 239
At a Glance 239
AS-i V2 Operating Mode: General. 240
AS-i V2 protégé Mode 242
AS-i V2 Offline Operating Mode 243

AS-i V2 Data Exchange Off Operating Mode. 244

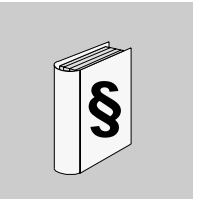
Chapter 16 AS-i V2 performance 245
Performances of the AS-i V2 bus. 245

Chapter 17 DFB for AS-i Security Monitor 247
At a Glance 247
Description of safety function block of the As-i bus 248
Operation of the AS-i security function block. 251
Security DFB programming rules. 252
How to implement a DFB function block 254

Chapter 18 Language objects of the AS-i V2 bus. 257
At a glance. 257
18.1 Language objects and IODDT's of the AS-i V2 communication 258
Presentation of language objects of the AS-i V2 bus. 258
18.2 Language objects and generic IODDT for all the communication protocols . . . 259
At a Glance 259
Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type 260
Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type 261
18.3 Language objects of the AS-i V2 function 263
At a Glance 263
Details of the implicit exchange objects of T_COM_ASI_V2 type IODDTs 264
Details of the explicit exchange objects of the T_COM_ASI_V2 type IODDT . . 269
Details of implicit exchange objects of the AS-i V2 function 271
Details of explicit exchange objects of the AS-i V2 function 274
Details of the objects managing the mode of operation for the AS-i V2 function 277
Details of configuration objects for the AS-i V2 function 278
18.4 IODDT T_GEN_MOD applicable to all the modules 279
Details of the language objects of the IODDT of the type T_GEN_MOD 279

Index 281

Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

DANGER

DANGER indicates an imminently hazardous situation, which, if not avoided, **will result** in death, serious injury, or equipment damage.

WARNING

WARNING indicates a potentially hazardous situation, which, if not avoided, **can result** in death, serious injury, or equipment damage.

CAUTION

CAUTION indicates a potentially hazardous situation, which, if not avoided, **can result** in injury or equipment damage.

PLEASE NOTE

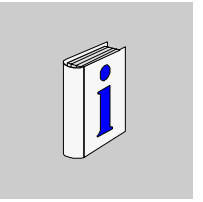
All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When controllers are used for applications with technical safety requirements, please follow the relevant instructions.

No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material. This document is not intended as an instruction manual for untrained persons.

© Schneider Electric 2005 All rights reserved.

About the Book



At a Glance

Document Scope

This manual describes hardware and software installation procedures for the AS-i bus.

Validity Note

The data and illustrations found in this documentation are not binding. We reserve the right to modify our products in line with our policy of continuous product development.

The information in this document is subject to change without notice and should not be construed as a commitment by Schneider Electric.

Product Related Warnings

Schneider Electric assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product.

For reasons of safety and to ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When controllers are used for applications with technical safety requirements, please follow the relevant instructions.

Failure to observe this product related warning can result in injury or equipment damage.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techpub@schneider-electric.com

The AS-i Bus

I

At a Glance

In This Chapter

This part introduces the AS-i bus on Premium and Atrium PLCs.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
1	General introduction to the AS-i bus	15

General introduction to the AS-i bus

1

At a Glance

Aim of this Chapter

This chapter provides a general introduction to the AS-i bus.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Reminder relating to AS-i and AS-i V2 buses	16
Overview of AS-i products from the Schneider catalog	17
Presentation of main constituent components	18
Example of AS-i bus topology	22
Main characteristics of the AS-i bus	23
Main Characteristics of the AS-i V2 Bus	25

Reminder relating to AS-i and AS-i V2 buses

General

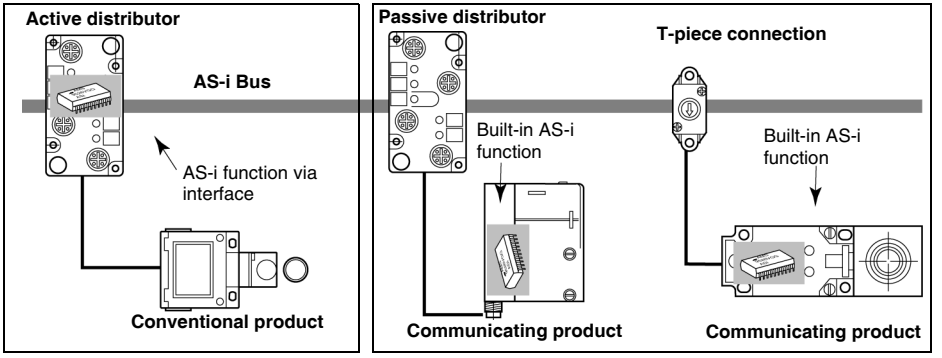
AS-i and AS-i V2 buses are field buses (level 0) and can be used to connect sensors/actuators. They allow "Discrete" type information to be routed between a bus master and sensor/actuator slaves. The AS-i V2 bus also allows for routing of analog information.
AS-i and AS-i V2 buses are made up of three major basic components:

- A dedicated power supply providing 30 V DC voltage.
- A bus master.
- Slaves (sensors, actuators and others).

The main types of sensors/ actuators

1. **Communication sensors/actuators:**
   With a built-in AS-i function they link up directly to the AS-i bus via a passive distributor or a T-piece connection.
2. **Conventional and IP65 sensors/actuators:**
   They connect to the bus via an AS-i interface (active distributor or Telefast IP20 discrete input-output bus interface). These interfaces are used to connect conventional sensors and actuators to the AS-i bus and provide them with capacity for dialog on the bus.

Illustration:



AS-i 124 Bus maximum of 124 products (124 inputs and 124 outputs) AS-i V2 Bus maximum of 434 I/Os (248 inputs and 186 outputs)	**AS-i Bus 31 products (sensors or actuators)** **AS-i V2 Bus maximum of 62 slaves (maximum of 31 standard or 62 extended slaves)**

AS-i Bus

You can only connect standard slaves to the AS-i bus.

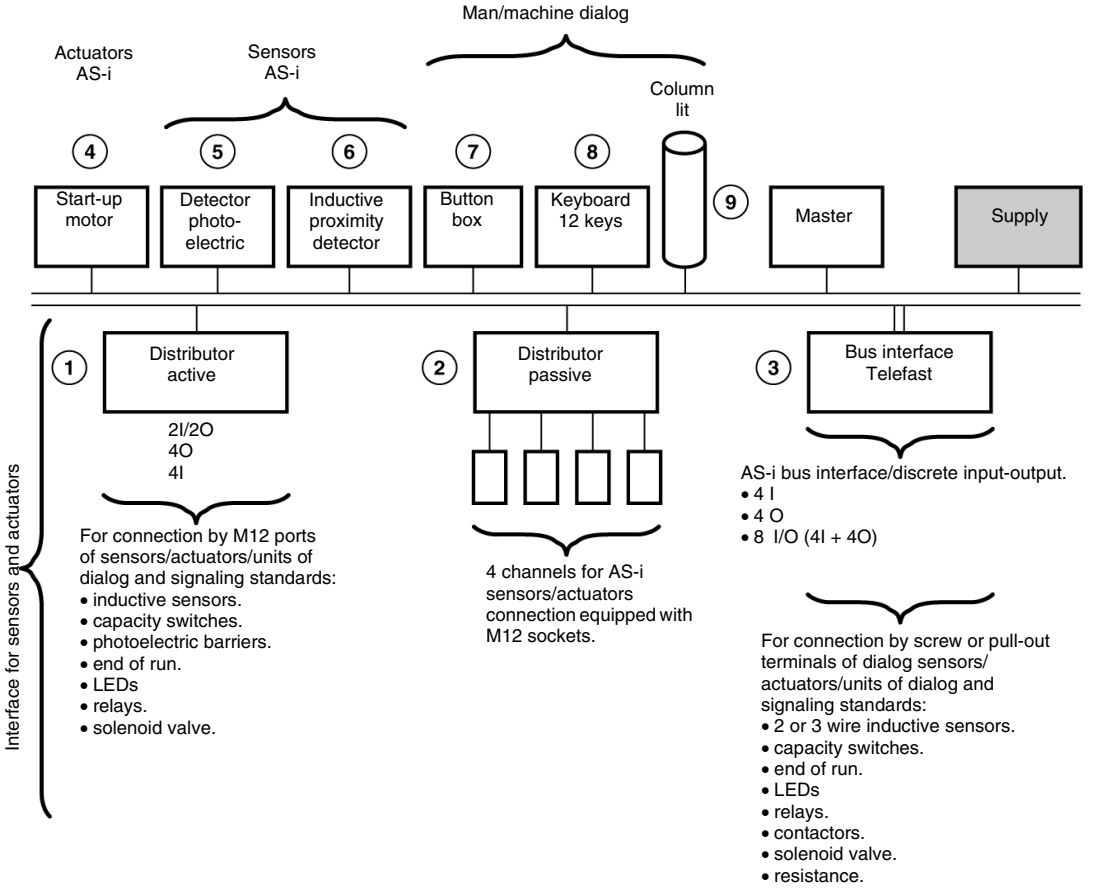
AS-i V2 Bus

You can connect standard slaves and extended slaves to the AS-i V2 bus.

Overview of AS-i products from the Schneider catalog

General

Non-exhaustive list of AS-i products from the Schneider catalog:



Presentation of main constituent components

Cable

This transmits data and carries the power. It can be made up from:

- either an unshielded, polarized twin-wire AS-i ribbon cable.
- Or a standard round, shielded or unshielded twin-wire cable.

Illustration:

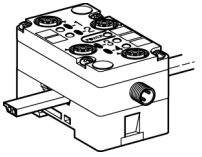
Polarized ribbon cable

Round cable

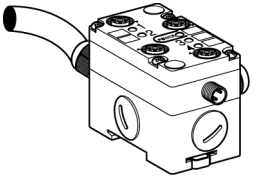
Active distributors

IP67 sealed interfaces for connecting sensors/actuators using M12 connectors. These distributors are used to connect "traditional", non-communicating sensors/actuators.

Illustration:



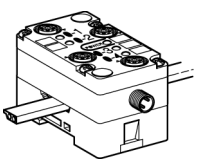
Active distributor for ribbon cable



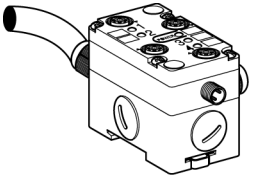
Active distributor for round cable

Passive distributors

IP67 sealed interfaces for connecting sensors/actuators using M12 connectors. These distributors do not have any electronics and can therefore be used to connect the "communicating" sensors/actuators.
Illustration:



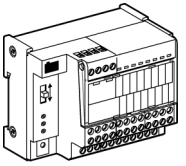
Passive distributor for ribbon cable



Passive distributor for round cable

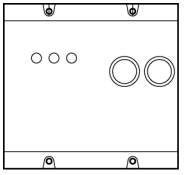
SB2 Telefast discrete inputs-outputs/bus interface

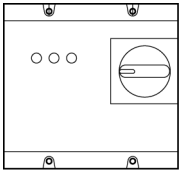
IP20 sealed interface with built-in AS-i function. It enables connection to all types of "traditional" non-communicating sensors/actuators via screw terminal blocks.
Illustration:



AS-i actuators

The direct motor starters and toggle switches in sealed boxes (IP54 and IP65) ensure electrical motors are controlled and protected up to 4KW at 400 V AC.
Illustration:

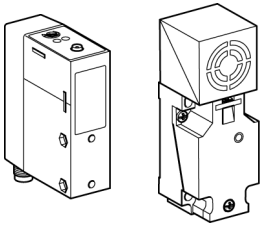




AS-i sensors

- Photo-electric detectors:
  They ensure that all kinds of objects (opaque, reflective etc.) are detected with 5 basic systems: barrier, reflex, polarized reflex, proximity and proximity with background blanked out. They offer an IP67 protection level.
- Inductive proximity detectors:
  They detect all metal objects and provide information for the functions used to check whether an object is present or not. They offer an IP67 protection level.

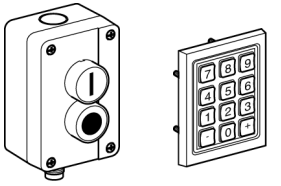
Illustration:



Man-machine interface products

- Button boxes:
  These are made up of dialog tools, which are perfectly adapted to exchanging information between the operator and machine.
- Keyboards:
  Man/machine dialog tools, these have 12 touch sensitive keys. The information delivered is coded in BCD on 4 bits. They offer an IP65 protection level.

Illustration:

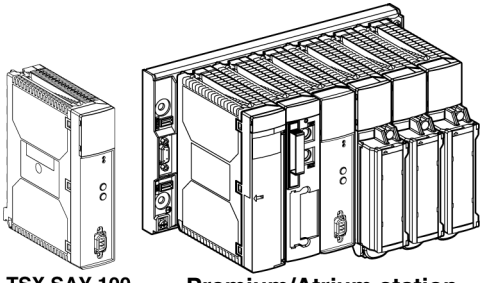


Signaling elements

- Illuminated indicator banks:
  Optical or sound signaling elements.

Bus Master

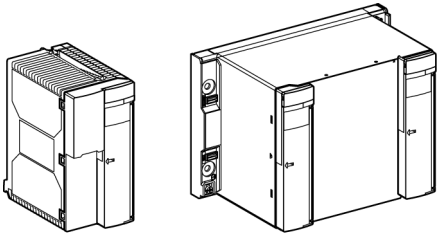
The AS-i bus master manages all data exchanges on the AS-i bus. In a Premium/Atrium PLC station, it is possible to integrate the **TSX SAY 100** module for the AS-i bus or the **TSX SAY 1000** module for the AS-i V2 bus.
Illustration:



TSX SAY 100 or TSX SAY 1000 **Premium/Atrium station**

AS-i Power Supply

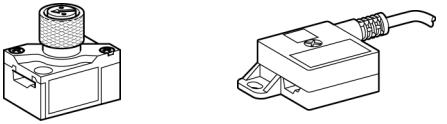
AS-i dedicated power supply, designed to supply the components connected to the AS-i bus.
The distribution of this power supply uses the same medium as that used for data exchange.
Illustration:



TSX SUP A02 **TSX SUP A05**

Connecting and branching accessories

T-piece connectors are used to make connections to the AS-i bus. These are designed to connect to AS-i ribbon cables or to ribbon/round cable branches.
Illustration:



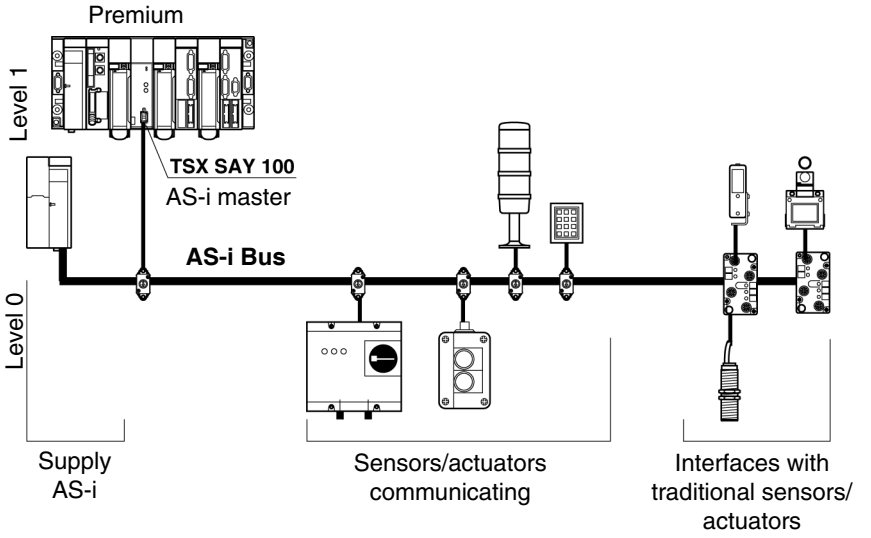
T-piece for ribbon cable

Ribbon/Round cable branch

Example of AS-i bus topology

General

Illustration:



Main characteristics of the AS-i bus

General

AS-i is a system in which exchange management is ensured by a single master who calls in succession, by scanning the bus, each detected slave and awaits a response.
The serial communication frame transmits:

- 4 data bits (D0 to D3), which are the image of inputs or outputs according to the nature of the interface.
- 4 parametering bits (P0 to P3), which are used to set the operating modes of the interface.
  P0 to P3 bits are used for "intelligent" devices, including AS-i ASIC (specific integrated circuit). Operation can be modified while it is running.

The address of the slave is coded on 5 bits.
At the request of the AS-i master, outputs are set and the inputs for AS-i devices are sent in the slave's response.

Slave Addressing

Each slave connected to the AS-i bus must have an address between 1 and 31 (coding on 5 bits).
The slaves delivered from the factory have the address 0 (the address of the slave is memorized in a non-volatile format).
Addresses are programmed using a specialized addressing terminal.

Note: When replacing a faulty slave whose address has been set, the address of the slave to be replaced can be updated automatically.

Identification of Slaves

All slave devices connected to the AS-i bus are identified by:

- An I/O Code (input/output distribution code).
- An identification code, which completes the functional identification of the slave.

These identifications allow the AS-i master to recognize the configuration, which is present on the bus.
These different profiles have been developed by the AS-i Association. They are used to distinguish between input, output and mixed modules, "intelligent" device families, etc.

Maximum Number of Inputs/Outputs

An AS-i bus can support a maximum of 31 slaves.
Each slave can have a maximum of 4 inputs and/or 4 outputs.
This makes it possible to manage a maximum of 124 inputs + 124 outputs, or 240 discrete inputs/outputs when all active devices have 4 inputs and outputs.

AS-i Cable

The AS-i cable is a twin-wire link on which communications and power supply for the connected devices are transmitted.
The link does not need to be twisted.
The cross-section of wires can be from 2 x 0.75 mm^2, 2 x 1.5 mm^2 or 2 x 2.5 mm^2, according to the current consumed by the devices.

Topology and Maximum Length of AS-i Bus

The topology of the AS-i bus is flexible. It can be perfectly adapted to meet the user's needs (point to point, on line, tree structure etc.).
In every case, the total length of all the branches of the bus must not exceed 100 meters without a repeater.

AS-i Bus Cycle Time

This is the cycle time between slave(s) and the **TSX SAY 100** module.
The AS-i system always transmits information, which is the same length to each slave on the bus. The AS-i cycle time depends on the number of slaves connected to the bus.
In the presence of 31 functioning slaves, this time period will be a maximum of 5 ms.

Reliability, Flexibility

Reliable operation is ensured by the transmission process used (Manchester current and coding modulation). The master monitors the line supply voltage and the data sent. It detects transmission errors as well as slave failures, and sends the information to the PLC.
Exchanging or connecting a new slave during operation does not disturb communications between the master and the other slaves.

Main Characteristics of the AS-i V2 Bus

General

AS-i is a system in which exchange management is ensured by a single master which, by scanning the bus, calls each slave in succession and awaits a response.
Communication series frame for slaves with standard AS-i address settings:

- 4 data bits (D0 to D3), which are the image of inputs or outputs according to the type of interface,
- 4 parametering bits (P0 to P3), which are used to set the operating modes of the interface.

Communication series frame for slaves with extended address settings:

- 3 or 4 data bits, which are the image of inputs (4 bits, D0 to D3) or outputs (3 bits, D0 to D2) depending on the type of interface,
- 3 parametering bits (P0 to P2), which are used to set the operating modes of the interface.

All slave devices connected to the AS-i bus are identified by at least one "I/O Code" and one "ID code" which completes the functional identification of the slave.
Some slaves have an ID1 code, which defines the internal functions of the slave: on analog slaves, for example, ID1 shows the slave's analog channel number.

Note: The frame base is the same for analog slaves as it is for TOR slaves. There is ascending compatibility between AS-i and AS-i V2. This means that all slaves on the market are supported by SAY 1000.

Slave Addressing

Each slave connected to the AS-i bus must have an address which lies between 1 and 31, either with "Bank" /A, or with "Bank" /B for extended address settings.
The slaves delivered from the factory have the address 0 (the address of the slave is memorized in a non-volatile format).
Addresses are programmed using a specialized addressing terminal.

Note: When replacing a faulty slave whose address has been set, the address of the slave to be replaced can be updated automatically.

Identification of Slaves

All **standard address setting slaves** connected to the AS-i bus are identified by:
- An I/O Code (input/output distribution code),
- An identification code, which completes the functional identification of the slave.

All **extended address setting slaves** connected to the AS-i bus are identified by:
- An I/O Code (input/output distribution code),
- An identification code which completes the functional identification of the slave.
- An ID1 code which defines the internal functions of the slave,
- An identification code (ID2) which completes the functional identification of the slave.

These identifications allow the AS-i master to recognize the configuration, which is present on the bus.
These different profiles have been developed by the AS-i Association. They are used to distinguish between input, output and mixed modules, "intelligent" device families, etc.

Maximum Number of Inputs/Outputs

On the same bus, an AS-i bus can support a maximum of:
- 31 4I and/or 4O standard address setting slaves, with addresses from 1 to 31
- 62 extended address setting slaves with 4I and/or 3O , using addresses from 1 A/B to 31 A/B.

This makes it possible to manage a maximum of 248 inputs +186 outputs (thus 434 inputs/outputs) when all extended slaves have 4 inputs and 3 outputs.

AS-i Cable

The AS-i cable is a twin-wire link on which communications and power supply for the connected devises are transmitted.
The link does not need to be twisted.

The cross-section of wires can be from 2 x 0.75 mm^2, 2 x 1.5 mm^2 or 2 x 2.5 mm^2, according to the current consumed by the devices.

Topology and Maximum Length of AS-i Bus

The topology of the AS-i bus is flexible. It can be perfectly adapted to meet the user's needs (point to point, on line, tree structure etc.).
In every case, the total length of all the branches of the bus must not exceed 100 meters without a relay.

AS-i Bus Cycle Time

This is the cycle time between slave(s) and the **TSX SAY 1000** module.
The AS-i system always transmits information, which is the same length to each slave on the bus. The AS-i cycle time depends upon the number of slaves connected to the bus in the presence of functioning slaves.

The scan time **t** represents the exchange time between a master and **n** active slaves (a maximum of 31 on /A or /B).
So, for:

- up to 19 active slaves, **t** = 3ms
- 20 to 31 active slaves t = (1+**n**) * 0.156ms
  When two slaves A and B have the same address, each slave in the pair is scanned every two cycles.
  This means that for 31 extended address setting slaves configured in /A, + 31 extended address setting slaves configured in /B. the scan time will be 10 ms.

Maximum cycle time:

- maximum 5 ms for 31 standard or extended address setting slaves,
- maximum 10 ms for 62 extended address setting slaves.

Reliability, Flexibility

Reliable operation is ensured by the transmission process used (Manchester current and coding modulation). The master monitors the line supply voltage and the data sent. It detects transmission errors as well as slave failures, and sends the information to the PLC.
Exchanging or connecting a new slave during operation does not disturb communications between the master and the other slaves.

Hardware installation for AS-i power supply units and TSX SAY 100 and TSX SAY 1000 AS-i bus interface modules

II

At a Glance

In This Chapter

This part deals with the hardware installation of AS-i power supply units, and **TSX SAY 100** and **TSX SAY 1000** AS-i bus master interface modules.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
2	AS-i bus power supply units	31
3	AS-i bus interface module: TSX SAY 100	59
4	AS-i V2 bus interface module: TSX SAY 1000	83

AS-i bus power supply units

2

At a Glance

Aim of this Chapter

This chapter introduces AS-i bus power supply units.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
2.1	Introduction to AS-i Bus power supply units	32
2.2	Installation of the AS-i bus power supply units	39
2.3	AS-i power supply connections	45
2.4	Characteristics of AS-i bus power supply units	54

2.1 Introduction to AS-i Bus power supply units

At a Glance

Aim of this Section

This section introduces AS-i bus power supply units.

What's in this Section?

This section contains the following topics:

Topic	Page
General introduction to AS-i power supply units	33
Physical description of the TSX SUP A02 power supply module	34
Physical description of the TSX SUP A05 power supply block	35
Physical description of the support board	36
Catalog of AS-i power supply units	37
AS-i power supply: specific features	38

General introduction to AS-i power supply units

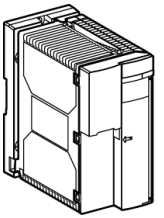
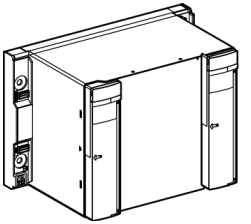
General

The AS-i power supply blocks and modules **TSX SUP A02** and **TSX SUP A05** are designed to provide a 30V DC power supply to devices connected to an AS-i field bus.
This power supply is distributed over the same wires as those used for data exchange.
The fixing system for these products has been specially designed to meet the particular mounting distance and fixing specifications for Premium PLCs and TBX products.

All the products are mounted:
- on a Telequick AM1-PA board,
- on a central AM1-DP200/DE200 DIN rail.

Presentation table

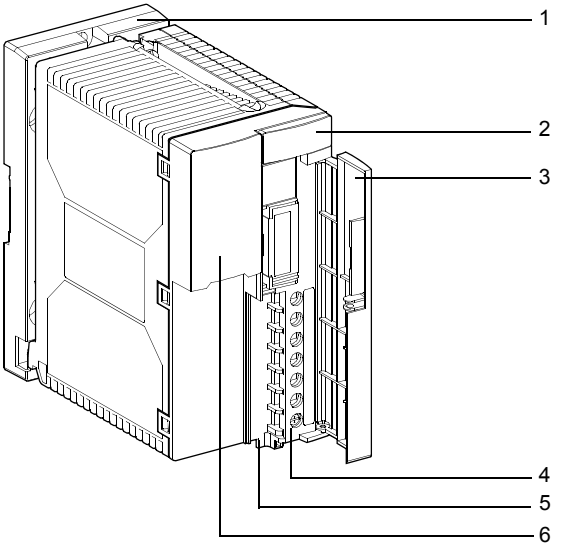
This table presents the different AS-i power supplies:

AS-i Bus power supply	
100...120 V AC or 200...240 V AC network voltage	
	
30 V DC AS-i / 2.4 A	**30V DC AS-i / 5A & 24V DC**

Physical description of the TSX SUP A02 power supply module

Illustration

Numbered diagram:



Reference table

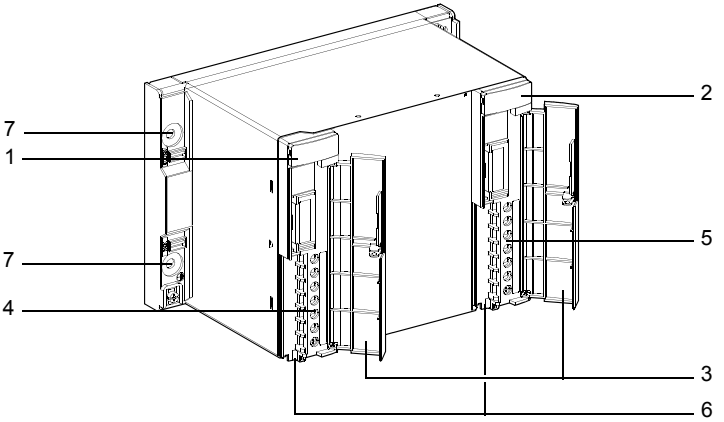
The following table provides descriptions for each of the numbers on the above diagram:

Numbers	Description
1	Support board enabling the power supply module to be fitted directly onto an AM1-DE200/DP200 type DIN rail or to a Telequick AM1-PA mounting plate
2	Display panel comprising: • an AS-i LED (green): lit if the internal and output voltages are correct.
3	Protective flap for terminal block.
4	Screw terminal block for connection: • to AC power supply network, • of the 30V DC AS-i output.
5	Hole for threading of cable tie.
6	110/220 V voltage selector switch. On delivery, the selector is set at 220V.

Physical description of the TSX SUP A05 power supply block

Illustration

Numbered diagram:



Reference table

The following table provides descriptions for each of the numbers on the above diagram:

Numbers	Description
1	Display panel with an ON LED (orange): lit when the power supply is switched on.
2	Display panel with: ● a 24V LED (green): lit when the 24V DC output voltage is present and correct, ● an AS-i LED (green): lit when the 30V DC output voltage is present and correct,
3	Protective flaps for terminal blocks.
4	Screw terminal block for connecting to the AC power supply network.
5	Screw terminal block for connecting the AS-i 24V DC and 30V DC outputs.
6	Hole for threading of cable tie.
7	Four screw holes for M6 screws.

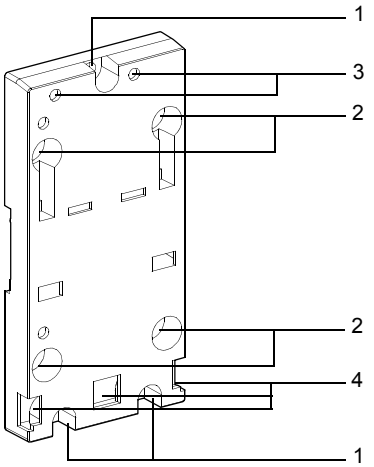
Physical description of the support board

Introduction

Each **TSX SUP A02** power supply module is supplied mounted on a supporting board that enables the power supply to be fixed: either onto a DIN AM1-DE200 or AM1-DP200 DIN mounting rail, or onto a Telequick AM1-PA perforated mounting plate.
Each support board can be fitted with one **TSX SUP A02** module.

Illustration

Numbered diagram:



Reference table

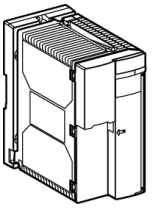
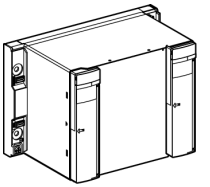
The following table provides descriptions for each of the numbers on the above diagram:

Numbers	Description
1	Three 5.5mm diameter holes enabling the support board to be fixed to a panel or to an AM1-PA perforated mounting plate with a mounting distance of 140mm (the standard mounting distance for Micro PLCs).
2	Four 6.5mm diameter holes enabling the support board to be fixed to a panel or to an AM1-PA perforated mounting plate with a mounting distance of 88.9mm (the standard mounting distance for Premium PLCs).
3	Two M4 screws enabling the TSX SUP A02 power supply to be fixed in place.
4	Slots for fixing the pins situated at the bottom and to the rear of the module.

Catalog of AS-i power supply units

Selection table

The following table lists the main characteristics of AS-i power supplies:

		
Input characteristics Nominal voltage	100...120 V AC or 200...240 V AC	
Threshold values	85..132 V AC or 170..264 V AC	
Frequency threshold	47..63Hz or 360..440Hz	
Nominal input current	1.3A	5A
Output characteristics Useful power	72W	230W
Output voltage (direct)	30 V DC AS-i	30V DC AS-i / 24V DC
Nominal current	**2.4A**	**5A(1)** / **7A(1)**
Auxiliary functions SELV safety (1)	Yes	
Parallelization (2)	No	
Redundancy (4)	No	Yes
Product references	**TBX SUP A02**	**TSX SUP A05**

(1) Maximum current for each output. The sum of the power values is limited to 230 W.
(2) Manufacturer's specifications in accordance with IEC 950, IEC 1131-2 standards guaranteeing user safety for the 24 V output, in terms of insulation between primary and secondary, maximum overvoltage on output wires, and protection via ground circuit.

AS-i power supply: specific features

General

The simultaneous transmission of information and power over the same cable requires that data transmission be filtered from the power supply.
This is the why the AS-i bus features a decoupling filter able to withstand the maximum direct current delivered by the power supply. The power supply has an impedance which is standardized relative to the data transmission frequencies.

2.2 Installation of the AS-i bus power supply units

At a Glance

Aim of this Section

This section describes the installation procedure for the AS-i bus power supply units.

What's in this Section?

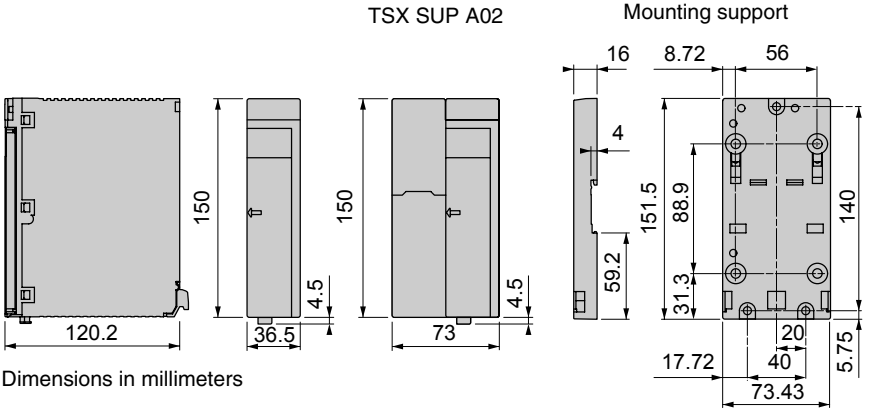
This section contains the following topics:

Topic	Page
Dimensions/mounting of the AS-i TSX SUP A02 power supply unit	40
Dimensions/mounting of the TSX SUP A05 AS-i power supply unit	43
Summary of methods of fixing	44

Dimensions/mounting of the AS-i TSX SUP A02 power supply unit

Dimensions

Illustration:

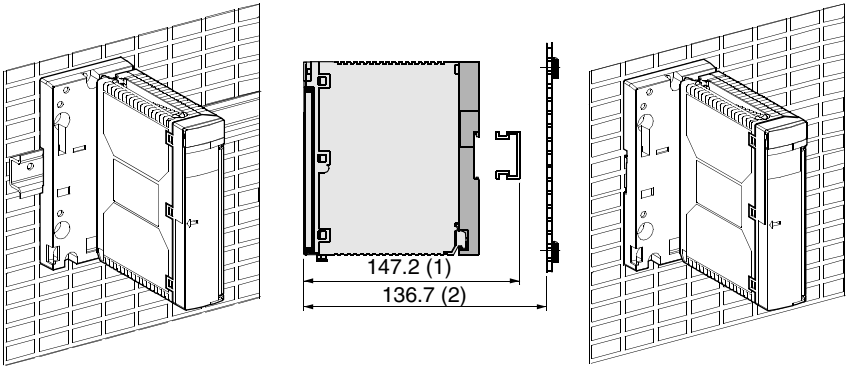


Dimensions in millimeters

Mounting on an AM1-DE200 or AM1-DP200 mounting rail or on an AM1-PA mounting plate

Each power supply module is supplied mounted on a support enabling this type of mounting.

Illustration:



(1) 147.2 mm (AM1-DE200), 139.7 mm (AM1-DP200)
(2) 136.7 mm (AM1-PA)

Mounting on an AM1-D•200 mounting rail

Carry out the following steps:

Step	Action
1	Check that the module is mounted on the support.
2	Mount the module + support on the mounting rail.

Mounting on an AM1-PA mounting plate

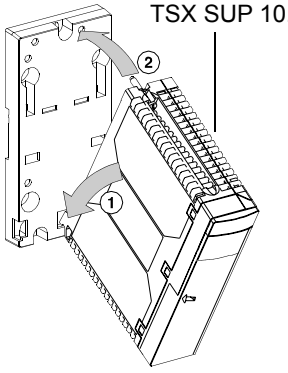
Carry out the following steps:

Step	Action
1	Remove the module from its support.
2	Mount the support on the AM1-PA mounting plate.
3	Mount the module on its support.

Mounting the module on the support

Each power supply module comes as standard with a support which allows it to be mounted directly on a DIN mounting rail. 1 TSX SUP A02 power supply module can be fitted to the support.

Carry out the following steps:

Step	Action	Illustration
1	Position the pins situated at the rear of the module (on the inside) in the corresponding slots of the support.	Steps 1 and 2 
2	Pivot the module upwards to bring it into contact with the support.	

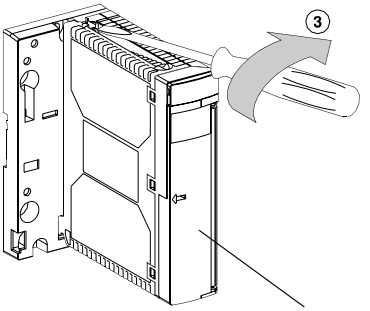
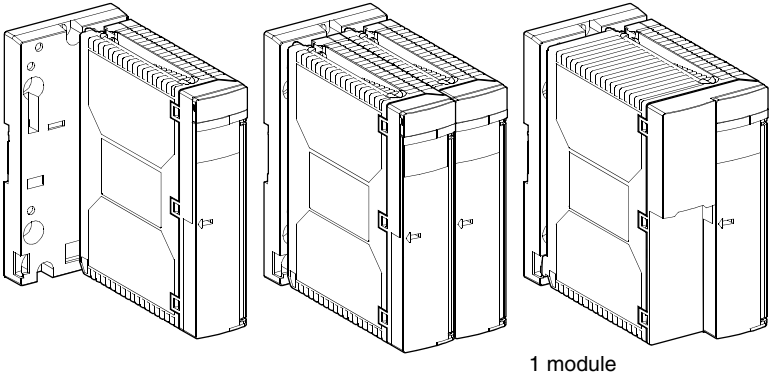
Step	Action	Illustration
3	Tighten the screw situated on the upper part of the module in order to fix it to the support.	Step 3  TSX SUP 10.1/A02

Illustration:



1 module
TSX SUPA02

Mounting on TSX RKY... rack

The TSX SUP A02 power supply module can be mounted in any position on the TSX RKY... rack, with the exception of the PS position reserved for the power supply module of the rack. In this case, the support is not used and should be removed.

These modules are mounted using the same procedure as for processor (See Premium and Atrium using Unity Pro Manual, Processors, racks and power supply modules,How to mount processor modules) modules.

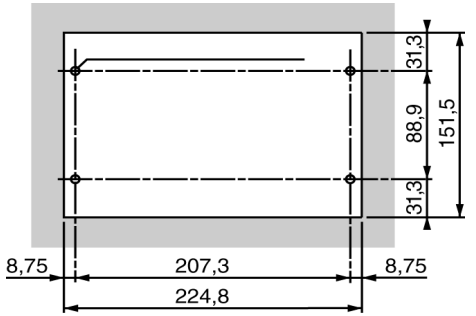
Dimensions/mounting of the TSX SUP A05 AS-i power supply unit

Introduction

The TSX SUP A05 power supply block can be mounted on a panel, an AM1-PA mounting plate or a DIN rail.

Mounting on a panel

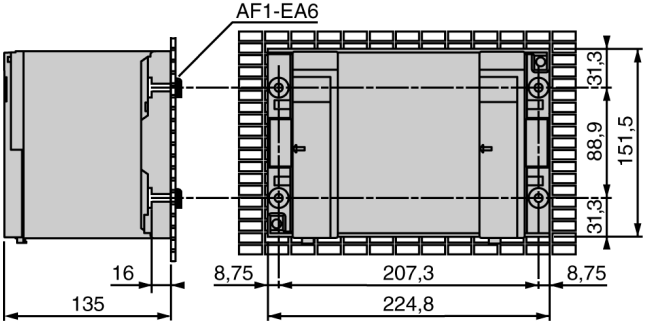
Drilling diagram (dimensions in millimeters):



(1) The diameter of the screw holes must be suitable for M6 screws.

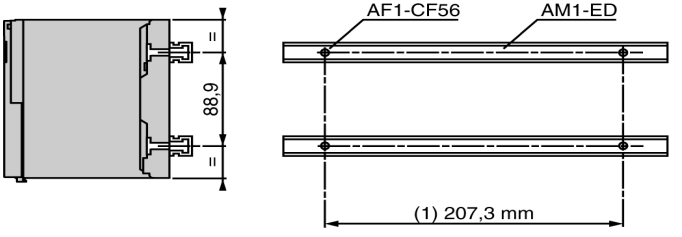
Mounting on a Telequick AM1-PA perforated mounting plate

Fix the power supply block in place using M6x25 screws + washers and AF1-EA6 clip nuts (dimensions in millimeters):



Mounting on a 35mm wide DIN mounting rail

Fix the power supply block in place using 4 M6x25 screws and AF1-CF56 1/4 turn sliding nuts (dimensions in millimeters):



Summary of methods of fixing

Summary table of fixing methods

The following table provides a summary of the different fixing methods available for AS-i power supply units:

Power supply reference	**TSX SUP A02**	**TSX SUP A05**
Telequick AM1-PA mounting plate	X	X
AM1-DE200/DP200 central DIN rail	X	-
AM1-ED DIN rail 140mm mounting distance (TSX 37 PLC)	X	-
AM1-ED DIN rail 88.9 mm mounting distance (Premium PLC)	X	X
Premium TSX RKY.. rack	X	-
Key	X: available - : unavailable	

2.3 AS-i power supply connections

At a Glance

Aim of this Section

This section outlines the principles to be followed when making AS-i power supply connections

What's in this Section?

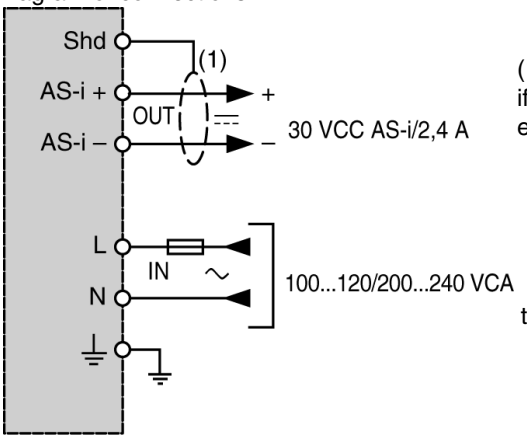
This section contains the following topics:

Topic	Page
TSX SUP A02 power supply connections	46
TSX SUP A05 power supply connections	48
General precautions	52

TSX SUP A02 power supply connections

Illustration

Diagram of connections:

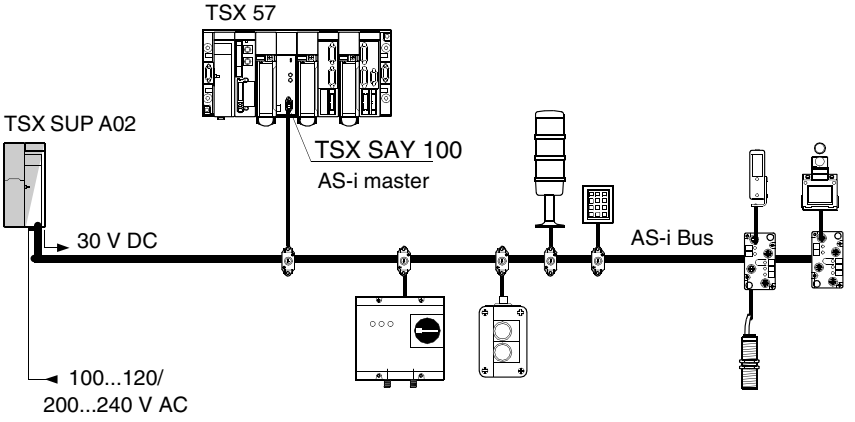


(1) AS-i shielded cable screen if in disturbed (EMC) environment.

Fu=External protection fuse on phase (Fu): 4A 250V time delay.

Synoptic of connections

The **TSX SUP A02** power supply unit is designed to supply the AS-i bus, and the connected slaves (connected to 30V DC/2.4A output).



Connection rules

Primary: observe wiring requirements for the phase and the neutral when wiring.

WARNING

To switch off the power supply to the AS-i bus, we recommend you fit a POWER OFF device to the primary power supply to the AS-i. This method will let you switch off all components of the AS-i bus (power supply, bus master and AS-i slaves).

Failure to follow this precaution can result in death, serious injury, or equipment damage.

DANGER

Safety of personnel

To ensure the safety of personnel, it is essential to connect the ground terminal of the module to the protective earth using a green/yellow wire.

Failure to follow this precaution will result in death, serious injury, or equipment damage.

To comply with isolation requirements for a 24 V SELV isolated voltage, the following wiring is used:

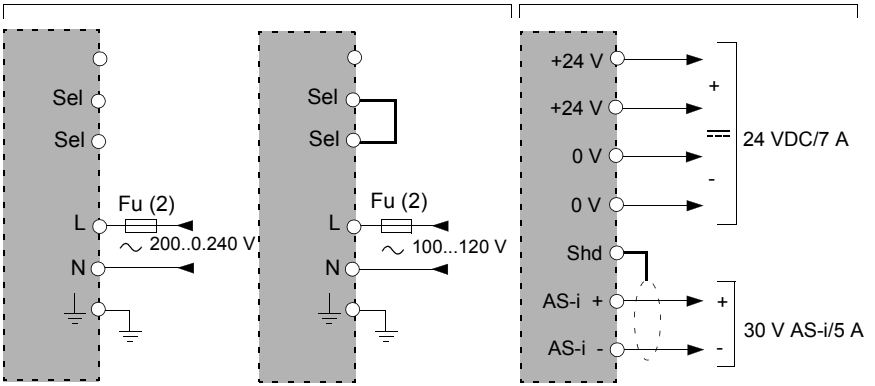
- an operating voltage >= 600V AC with a 1.5mm cross-section^2 for the mains connection,
- an operating voltage >= 300V AC with a cross-section of 25mm^2 for the 24 V outputs and the ground.

It is necessary to use a shielded cable for the AS-i bus only in cases where the installation is subject to very high levels of disturbance in terms of EMC (Electro Magnetic Compatibility).

TSX SUP A05 power supply connections

Illustration

Diagram of connections:

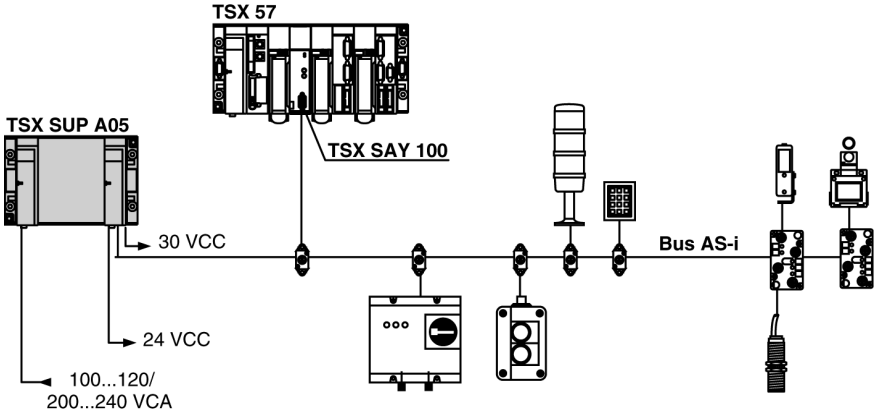


(1) Connection to be made when using a 100...120V AC power supply network.
(2) External protection fuse on phase (Fu): 6.3A 250V time delay.
(3) AS-i shielded cable screen if in disturbed (EMC) environment.

Synoptic of connections

The **TSX SUP A05** power supply unit is designed to supply the AS-i bus, and the connected slaves (connected to 30 V/5A output). It is also equipped with an auxiliary 24 VDC/7A power supply for high power consumption sensors/actuators; this is to be connected using the black AS-i ribbon cable.

Principle diagram:



Connection rules **Primary:** observe wiring requirements for the phase and the neutral when wiring.

WARNING

To switch off the power supply to the AS-i bus, we recommend you fit a POWER OFF device to the primary power supply to the AS-i. This method will let you switch off all components of the AS-i bus (power supply, bus master and AS-i slaves).

Failure to follow this precaution can result in death, serious injury, or equipment damage.

DANGER

Safety of personnel

To ensure the safety of personnel, it is essential to connect the ground terminal of the module to the protective earth using a green/yellow wire.

Failure to follow this precaution will result in death, serious injury, or equipment damage.

The "AC power supply network" terminal and 24V and 30 V DC AS-i terminals are protected by a flap allowing access to the wiring terminals. The wiring outputs descend vertically, and can be held in place by a cable tie.

Secondary:
To comply with isolation requirements for a 24 V SELV isolated voltage, the following wiring is used:

- an operating voltage >= 300V AC with a 1.5mm^2 or 2.5mm^2 cross-section for connection to the mains,
- an operating voltage >= 300V AC with a cross-section of 2.5mm^2 for the 24 V outputs and the ground.
- wire the two 24V terminals in parallel or distribute the loads between the two 24V outputs when the total current to be delivered is greater than 5A.

It is necessary to use a shielded cable for the AS-i bus only in cases where the installation is subject to very high levels of disturbance in terms of EMC (Electro Magnetic Compatibility).

Given the high level of current this power supply can generate, its position on the bus is of great importance. If the power supply is positioned at one end of the bus, it provides its nominal current (for example 5A) for the entire bus, and the voltage drop at the other end of the bus is therefore proportional to the 5A delivered. If it is positioned in the middle of the bus, the voltage drop at either end is only proportional to 2.5A where the power consumption on each of the two sections of the bus is equal.

If there is no slave with high power consumption, it is advisable to position the power supply at the center of the installation. Conversely, if the installation has one or more high-power consumers, it is advisable to position the power supply close to these devices.

Note: Where high-power consumption actuators (contactors, solenoid coils , etc.), the TSX SUP A05 power supply can provide the auxiliary 24 V DC, isolated from the AS-i line.

General precautions

Introduction

It is essential when installing the yellow AS-i cable to place it in a separate cableway to the power cables. It is also preferable to place it flat and to avoid twisting in order to optimize the symmetry between the two wires of the AS-i cable.
The installation of the AS-i cable on a surface connected to the electrical potential of the machine (the frame for example), is recommended in order to comply with the requirements of the EMC (Electro Magnetic Compatibility) directive.

The cable end (or ends for a bus in star configuration) must be protected by connecting it to a T-piece or by ensuring that it does not protrude from the last connection point.

Important

It is essential to distribute the power evenly over the AS-i bus so that each accessory connected to the bus is supplied with sufficient voltage to ensure that it operates correctly. To achieve this, it is necessary to observe the following rules.

Rule 1

Choose a power supply with a rating suited to the total power consum
ption of the AS-i segment. The available are ratings 2.4 A (TSX SUP A02) and 5A (TSX SUP A05).
A rating of 2.4 A is generally sufficient for an average power consumption of 65 mA per slave for a segment comprising a maximum of 31 slaves.

Rule 2

To minimize the effects of voltage drops and to reduce cable costs, it is necessary to determine the optimum power supply position on the bus, and also the minimum cable cross-section appropriate for the distribution of power.
The voltage drop should not exceed 3V between the master and the last slave on the bus. The table below lists the elements necessary for choosing the cross section of the AS-i cable.

Table of characteristics:

AS-i cable cross-section	**0.75 mm^2**	**1.5 mm^2**	**2.5 mm^2**
Linear resistance	52 milli Ohms/meter	27 milli Ohms/ meter	16 milli Ohms/ meter
Voltage drop for 1A over 100 meters	5.2V	2.7V	1.6V

The cable that meets the requirements for the majority of applications is the cable with a 1.5 mm^2 cross-section. It is the standard model for the AS-i bus (cable proposed by the SCHNEIDER catalog).
Cables of smaller cross-section can be used where the sensors consume very little power.

Note: The maximum length without repeaters of all the segments making up the AS-i bus is 100 meters. The lengths of cable connecting a slave to a passive distributor must be taken into account.

2.4 Characteristics of AS-i bus power supply units

At a Glance

Aim of this Section

This section describes the electrical characteristics of AS-i bus power supply units.

What's in this Section?

This section contains the following topics:

Topic	Page
Electrical characteristics of AS-i power supply units TSX SUP A02/A05	55
Physical characteristics of environment	57

Electrical characteristics of AS-i power supply units TSX SUP A02/A05

Table of characteristics

The following table describes the electrical characteristics of the power supply units: TSX SUP A02/A05

AS-i power supply			**TSX SUP A02 30V/2.4A AS-i**	**TSX SUP A05 24V/7A AS-i & 30V/5A AS-i**	
Primary:					
Nominal input voltage		V	AC 100...120/200...240	AC 100...120/200...240	
Threshold input voltage		V	AC 85...132/170...264	AC 85...132/170...264	
Network frequency		Hz	47...63/360...440	47...63/360...440	
Nominal input current (U=100V)		A	1.3	5	
Maximum inrush current (1)	at 100 V	A	30	50	
	at 240 V	A	30	50	
Maximum it upon activation (1)	at 100 V	As	0.06	0.17	
	at 240 V	As	0.03	0.17	
Maximum i^2t upon activation (1)	at 100 V	A^2s	4	8.5	
	at 240 V	A^2s	4	8.5	
Power factor			0.6	0.51	
Harmonic 3			10% (Phi=0°and 180°)	10% (Phi=0°and 180°)	
Efficiency at full load		%	>75	>80	
Secondary:					
Useful power		W	72(84) (2)	230 (3)	
Peak nominal current	30 V AS-i output 24 V output	A A	2.4(2.8) (2) -	5 (3)(4) 7 (3)(4)	
Output voltage Global variation (-10°C to +60°C)		V V	30(AS-i) 29.5 to 31.6	24 +/-3%	30(AS-i) 29.5 to 31.6
Ripple (from 10 to 500kHz) Ripple (from 0 to 10kHz)		mV mV	50 300	200 240	50 300
Startup time on resistive load		s	<2(with C=15000 micro Farads	<2(with C= 15000 micro Farads	
Duration of mains power supply micro-cuts (5)		ms	<=10	<=10	

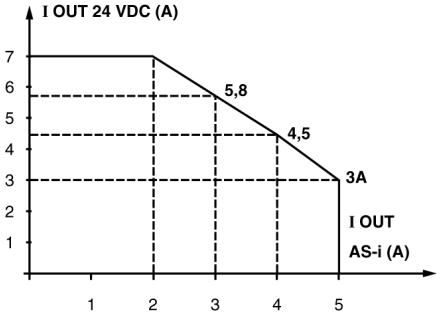
AS-i power supply			TSX SUP A02 30V/2.4A AS-i	TSX SUP A05 24V/7A AS-i & 30V/5A AS-i
Protection against	Short-circuits and overloads		fallback to 0 and automatic reactivation on disappearance of fault	limitation of current on each output
	Overvoltage	V	peak clipping U>36	peak clipping U>36
Power dissipation		W	24	60

(1) Values at initial power-up, at 25°C. These items must be taken into account at start-up for the dimensioning of protection devices.
(2) Useful power and output current for an ambient temperature of 60°C. Value between() = transient useful power.
(3) Useful power and output current for a maximum ambient temperature of 55°C, if product index II = 01. (60°C if product index II > 01).
(4) See diagram showing the distribution of current for each output, on the next page.
(5) Acceptable duration, at nominal voltage for a repetition period of 1Hz.

Diagram of currents available for 30 V AS-i outputs and 24 V outputs of the TSX SUP A05 power supply block

The maximum power delivered by the power supply is 230 W. If the consumption is 5 A on the 30 V AS-i, the possible load on the 24 V output is only 3 A (see diagram below).

Diagram:



Physical characteristics of environment

Table of characteristics

The following table describes the electrical characteristics of TSX SUP A02/A05 power supplies:

AS-i power supply module/block		**TSX SUP A02/A05**
Connection to screw terminals max. capacity per terminal	 mm^2	A02:1 output terminal A05: 2 terminals/output (24 V DC) and 1 terminal/output (30 V DC AS-i) 2 x 1.5 with cable end or 1 x 2.5
Temperature: Storage Operation	 °C °C	- 25 to +70 -10 to +60 (1)
Relative humidity	%	5-95
Cooling	%	By natural convection
User safety		SELV (EN 60950 and IEC1131-2)
Dielectric strength: Primary/secondary Primary/ground Secondary/ground	 V rms V rms V rms	 3500 2200 500
Insulation resistance Primary/secondary Primary/ground	 Mega Ohms Mega Ohms	 >=100 >=100
Leakage current		I<=3.5 mA (EN 60950)
Immunity to electrostatic discharges		6 kV on contact/8 kV in air (compliant with IEC 1000-4-2)
Electrical Fast Transient		2 kV (series mode and common mode commun on input and output)
Influence of electromagnetic field		10 V/m (80MHz at 1GHz)
Rejected electromagnetic disturbances		(compliant with FCC 15-A and EN 55022 class A Test conditions: nominal U and I, resistive load, cable: 1 meter horizontal, 0.8 meter vertical
Shock wave		Input: 4kV MC, 2kV MS Outputs: 2kV MF, 0.5 kV MS (compliant with IEC 1000-4-5)
Vibration (2)		1 mm 3 Hz at 13.2 Hz 2g 57 Hz at 150 Hz (compliant with IEC 68-2-6, FC test)

AS-i power supply module/block		**TSX SUP A02/A05**
Degree of protection		IP 20.5
MTBF at 40°C	H	100 000
Lifetime at 50°C	H	30 000 (at nominal voltage and at 80% of nominal power)

(1) -10°C +55°C for TSX SUP A02/A05 power supply with product index II=01.
(2) compliant with IEC 68-2-6, FC test with module or block mounted on plate or panel.

AS-i bus interface module: TSX SAY 100

3

At a Glance

Aim of this Chapter

This Chapter only deals with hardware installation of the **TSX SAY 100** interface module, AS-i bus master on a Premium/Atrium PLC.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
3.1	Description of the TSX SAY 100 Module	60
3.2	AS-i Bus diagnostics	72
3.3	Operating modes of the TSX SAY 100 module	79
3.4	Precautions of use	81

3.1 Description of the TSX SAY 100 Module

At a Glance

Aim of this Section

This Section deals with hardware installation and the characteristics of the **TSX SAY 100** module.

What's in this Section?

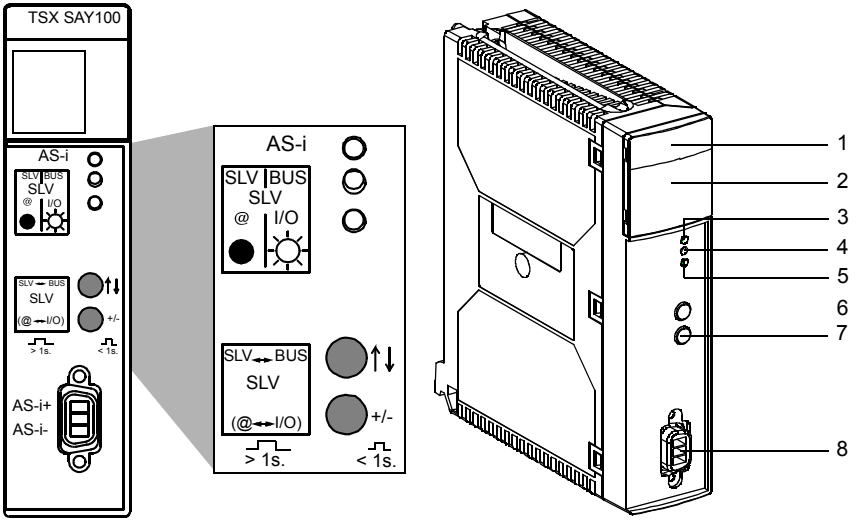
This section contains the following topics:

Topic	Page
Physical description	61
Mounting/installation	63
Connections	64
Displaying module states	67
Specific Display Panels on the TSX SAY 100 Module	68
Technical characteristics	70
Safety of personnel	71

Physical description

General

The TSX SAY 100 module comes as a standard format module.
Illustration:



Front panel

Reference table The following table describes the diagrams above using the numbers shown:

Number	Description
1	A display panel consisting of 4 LEDs for displaying the operating modes of the module: ● RUN LED (green): when lit, this indicates normal functioning of the module. ● ERR LED (red): when lit, this indicates a module error. ● COM LED (green): when lit, this indicates data exchanges on the AS-i network. ● I/O LED (red): when lit, this indicates an external input/output error on the AS-i bus.
2	Display panel consisting of 32 LEDs (0 to 31) which enable AS-i bus diagnostics, and display the state of each slave connected to the bus.
3	AS-i LED (red): when lit, this indicates an AS-i power supply fault.
4	Bus LED (green): when lit, this indicates that display panel**2** is in BUS display mode (display of slaves on the bus).
5	I/O LED (green): when lit, this indicates that display panel **2** is in Slave "SLV" display mode (displays status of input/output bits for a selected slave).
6	Push button "↑↓" designed for local diagnostics of the AS-i bus. You can move between the different AS-i bus diagnostic modes by pressing this button (long or short presses), combined with the "+/-" push button.
7	Push button "+/–" designed for local diagnostics of the AS-i bus. You can move between the different AS-i diagnostics modes by pressing this button (long or short presses), combined with the "↑↓" push button.
8	CANNON SUB D connector for connection to AS-i bus.

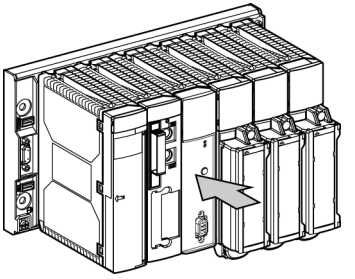
Mounting/installation

General

The **TSX SAY 100** module can be mounted in any position on the **TSX RKY** rack, except for those positions dedicated to processors and the power supply.

Note: In the event where the module is mounted on a remote X Bus rack, the maximum allowed distance from the processor must be 175 meters minus the length of the X Bus (max. 100 m).

The mounting and removal (See Premium and Atrium using Unity Pro Manual, Discrete I/O modules, Fitting of discrete IO modules) procedure is identical to the mounting and removal procedure for other modules.
Example of mounting a TSX SAY 100 module:



Note: The module can be mounted and removed with both PLC and AS-i bus power switched on.

Number of Modules per Station

Number of "application-specific" channels supported:

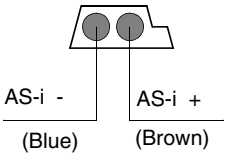
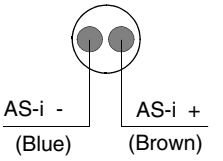
- Premium (See Premium and Atrium using Unity Pro Manual, Processors, racks and power supply modules, Catalog of TSX 57 Processors)
- Atrium (See Premium and Atrium using Unity Pro Manual, Processors, racks and power supply modules, Catalog of Atrium Processors)

Connections

AS-i Bus Cables

AS-i bus cables carry the signals and supply the sensors and actuators connected to the bus with 30 V DC.

Types of AS-i cables:

Cable type	Specifications	Illustration
Polarized AS-i ribbon cable	Color: yellow. Wire cross-section: 1.5 mm^2	
Standard round cable	Wire cross-section: 1.5 mm^2 or 2.5 mm^2	

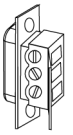
Recommended cable: product reference H05VV-F2x1.5, conforms to the DIN VDE 0281standard. Wire cross-section: 1.5 mm^2.

Cable Routing

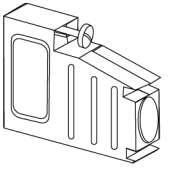
The AS-i cable and the power cables carrying higher power levels must be in separate ducts, which are protected by a metal screen.
When using a shared route for control cables it is essential that the connections on these control links should conform to the technology rules (e.g. the discharge diode or limiters on the terminals of self-inductive elements etc.).

Connector

A set (connector + cover) is delivered with the module, which is used to connect the module to the AS-i bus. This connector must be linked to the cable of the AS-i bus and assembled by the user according to the procedure described later.
Illustration:



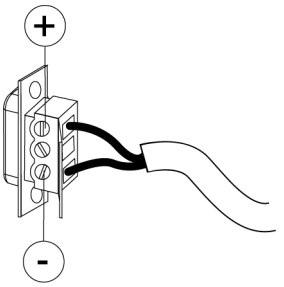
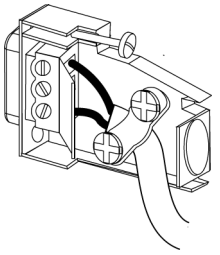
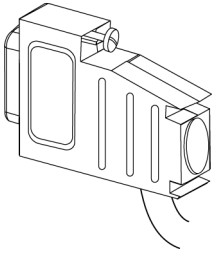
Connector

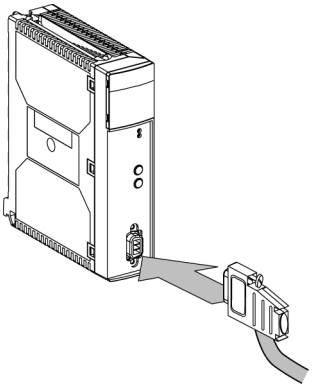


Cover

Connection of Module to Bus

To connect a module to the bus, follow the procedure below:

Step	Action
1	Connect the 2 wires of the AS-i cable to the connector, taking the polarities into account:  In the special event that a shielded cable is used, this should be connected to the central terminal.
2	Mount the connector in its cover and fix the cable to it: 
3	Click the cover shut: 

Step	Action
4	Mount the assembled unit on the module: 

Displaying module states

General

This is carried out with 4 LEDs RUN, ERR, COM, I/O which are located on the module. Their state (LED off, flashing or on) provides information on the operating mode of the module.

LED state:

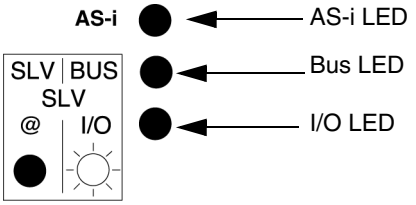
LEDs	**On**	**Flashing**	**Off**
RUN (green)	Module operating normally	Module self-testing (1)	Faulty module, or module switched off
ERR (red)	Serious internal fault, module broken down	Module self-testing (1) Fault: system OK but: ● application fault or ● AS-i bus error	No internal error
COM (green)	-	Module self-testing (1) Communication on AS-i bus	No communication on AS-i bus
I/O (red)	Input/output error	Module self-testing (1)	Module operating normally
(1) the 4 LEDs flash simultaneously during self-testing mode when the module is switched on.			

Specific Display Panels on the TSX SAY 100 Module

General

3 LEDs: AS-i, Bus and I/O display information specific to the TSX SAY 100 module.

View of the 3 LEDs:

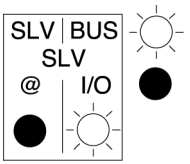
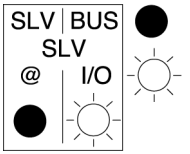
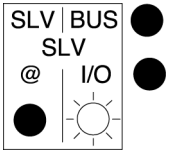


AS-i LED (red)

LED state	Meaning
LED on	Module operating normally
Flashing LED	Supply fault on the AS-i bus
LED off	Automatic addressing initialized

Bus and I/O LEDs These two LEDs display the view mode selected:

- Bus display mode or
- Slave display mode.

Display module	LED state	Meaning
	Bus LED on, I/O LED off	The 32 LED display panel located at the front of the module is in BUS display mode and displays all the slaves present on the bus.
	Bus LED off, I/O LED on	The 32 LED display panel located on the front of the module is in Slave display mode (SLV) with display of the input/ output status for a selected slave.
	Bus LED off, I/O LED off	The 32 LED display panel located on the front of the module is in Slave display mode (SLV) with display of the address of the slave selected.

Technical characteristics

AS-i Bus

Characteristic	Value
AS-i bus maximum cycle time	5ms
Maximum number of slaves on the AS-i bus	31
Maximum length of AS-i bus (all branches mixed without repeater)	100 meters
Maximum number of inputs/outputs	124 inputs +124 outputs
Nominal supply voltage for AS-i bus	30 V DC

TSX SAY 100 Module

Characteristic	Value
Programming the TSX SAY 100 module	using Unity Pro software
Response time with 31 slaves (1) for a PLC cycle time of 10 ms	Typically 27 ms, 37 ms maximum
Calculation of AS-i scanning time for n slaves (normal operation)	156 µs x (n+2) if n<31 156 µs x (n+1) if n=31
Power consumption 5V PLC	Typically 110 mA / 150 mA maximum
Power consumption 30 V AS-i	Typically 50 mA / 60 mA maximum
Power dissipation	2.5 W max.
Protection from polarity inversion on AS-i bus inputs	Yes
Degree of protection	IP20
Operating temperature	0 to 60 degrees Celsius
AS-i master profile	M2
Standards and service conditions	Compliant with those for Premium PLCs
(1) Logical response time + time between an AS-i input activated on the bus, processed in the PLC application and applied to an AS-i output.	

Safety of personnel

General

To ensure safety of personnel it is essential:

- To earth the ground terminal of the PLC.
- To use an AS-i SELV (Safety Extra Low Voltage) supply, nominal voltage 30 V DC.
- For PLCs which are connected to an alternating current network, a residual current circuit breaker must be placed upstream of this network, and this will cut off the PLC power supply source if ground leakage if detected.
- For PLCs which are connected to a direct current supply source you must ensure that the supply placed upstream of the PLC is SELV.
- To use certified AS-i products on the bus.

Due to the type of technology and connection used, the **TSX SAY 100** module can only run on 5 V DC and its 0V is linked to the PLC ground.

3.2 AS-i Bus diagnostics

At a Glance

Aim of this Section

This Section deals with the diagnostics mode carried out by the **TSX SAY 100** module.

What's in this Section?

This section contains the following topics:

Topic	Page
Introduction to AS-i Bus diagnostics	73
Moving between the different display modes	75
Displaying slaves on the AS-i bus	76
Displaying the state of the inputs/outputs of each slave	77

Introduction to AS-i Bus diagnostics

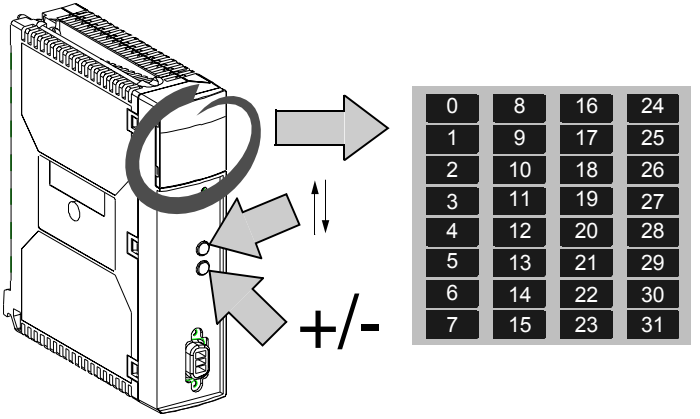
General

The module display panel is used for:

- Displaying the presence of each slave on the AS-i bus (Bus mode).
- Displaying the state of input/output bits of each slave present on the bus (Slave mode "SLV").

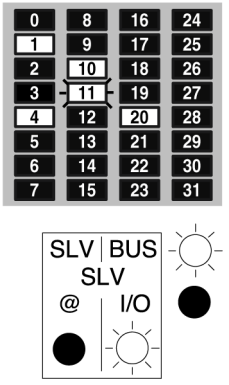
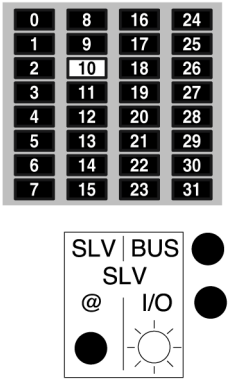
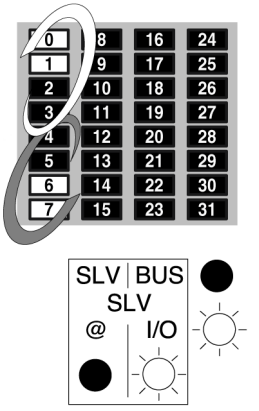
These modes can be accessed by a combination of actions on the push buttons (↑↓ and +/-) on the **TSX SAY 100** module.

Illustration:



Display modes

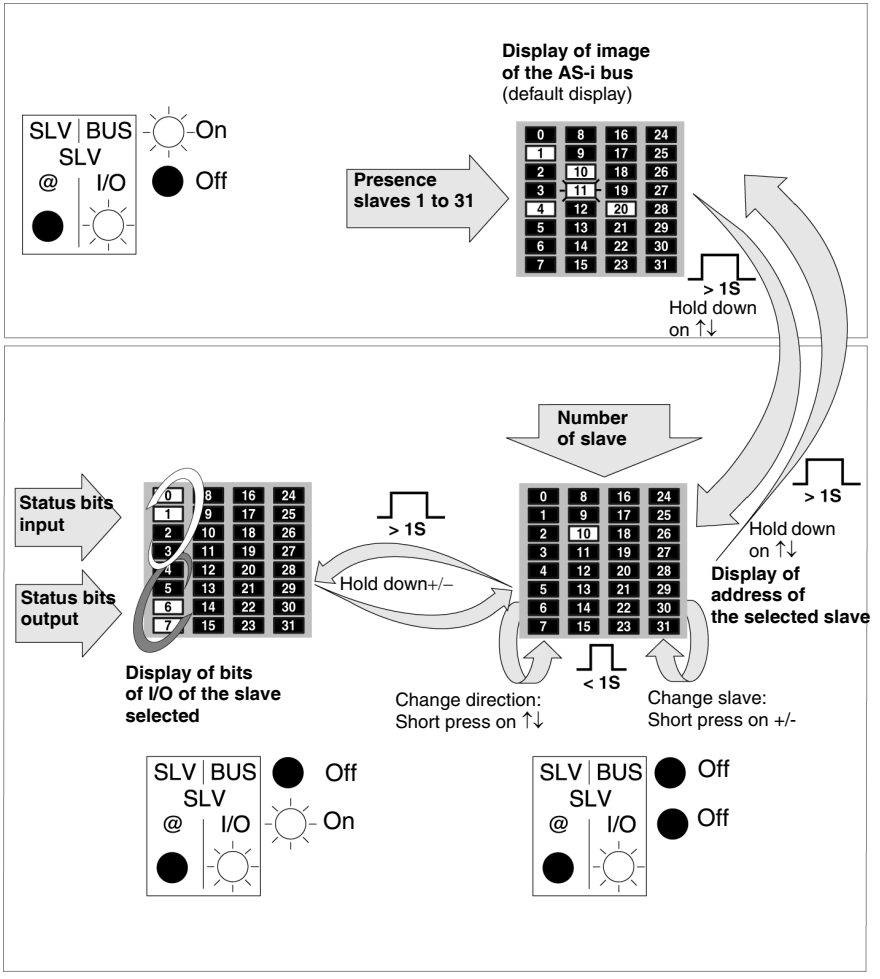
Table giving the 2 display modes of the module:

Bus mode	**Slave mode (SLV)**	
View of AS-i bus image. Each LED 1 to 31 corresponds to a slave address on the bus. ● LED on: slave present ● flashing LED: slave expected and not detected, or not expected and detected. ● LED off: slave not expected and not detected.	View of selected slave address. ● LED on: number of slave selected.	View of input/output bit state for slave selected. ● LEDs 0 to 3 display the state of the input bits. ● LEDs 4 to 7 display the state of the output bits. ● LED on: bit in state 1. ● LED off: bit in state 0 or not significant
Illustration:  The display mode is shown the Bus LED being on and the I/O LED being off.	Illustration:  The display mode is shown by Bus and I/O LEDs both being off.	Illustration:  The display mode is shown by the Bus LED being off and the I/O LED being on.

Moving between the different display modes

Illustration

This illustration shows how to move between the different display modes:



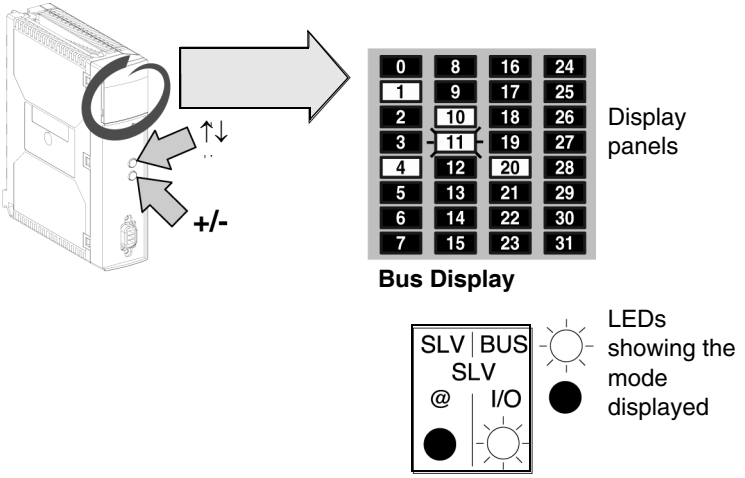
Displaying slaves on the AS-i bus

General

This mode is displayed by default when the module is switched on, and is used to display:

- The expected and detected slaves (LED permanently on).
- The unexpected and undetected slaves (LEDs off).
- The expected and undetected slaves (LEDs flashing).

Illustration:



The image of the AS-i network is displayed on the entire display panel. Each LED represents an AS-i bus slave address.
You can move between the different modes by a combination of actions on the push buttons ↑↓ and +/- : see *Moving between the different display modes, p. 75*.
Two LEDs "Bus" and "I/O" indicate the current display mode.

In the current example, the "BUS" LED is on and the "I/O" LED is off which indicates that the display is in Bus mode.
In the illustration above, the display panel indicates that:

- Slaves 1, 4, 10 and 20 (LEDs on) are present.
- Slave 11 (flashing LED) is present and not expected, or expected and missing.

Displaying the state of the inputs/outputs of each slave

General

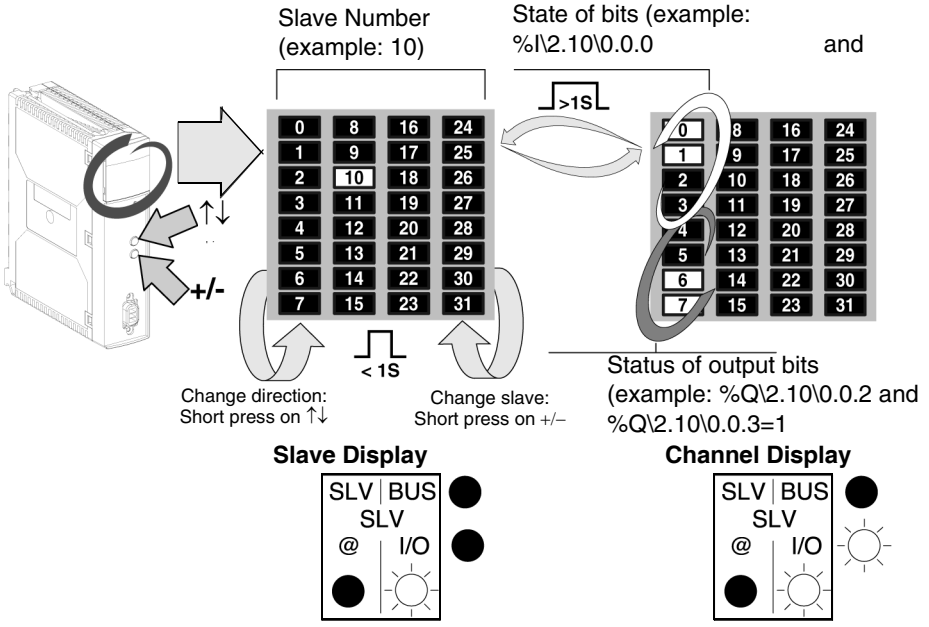
In this mode (Slave "SLV"), the display panel is used to view the state of input/output bits for each slave present on the bus.

Procedure to be followed

To access the state of input/output bits for a slave from BUS mode, proceed as follows:

Order	Action	Result
1	Press **for more than one second** on the button ↑↓	The display switches to Slave "SLV" mode.
2	Press **quickly** on the +/- button	The display of the slave address **goes up** from 1 to 31.
3	Press **quickly** on the button ↑↓	The direction of the slave address scan is reversed.
4	Press **quickly** on the +/- button	The display of the slave address **goes down** from 31 to 1.
5	Press**for more than one second** on the button +/–	• LEDs 0 to 3 display the state of input bits for the slave selected (1). • LEDs 4 to 7 display the state of output bits for the slave selected (1).
6	Press **quickly** on the +/- button	The state of inputs-outputs for the **following** slave is displayed.
7	Press **quickly** on the button ↑↓	The direction of the slave address scan is reversed.
8	Press **quickly** on the +/- button	The state of input-outputs for the **previous** slave is displayed.
9	Press**for more than one second** on the button +/–	The display shows again the number of the slave selected.
(1) **LED on = bit at 1** **LED off = bit at 0 or no input or output**		

Illustration



3.3 Operating modes of the TSX SAY 100 module

Operating modes of the TSX SAY 100 module

Output Fallback Position

The fallback mode is defined in the configuration and can be read from the word %KWr.m.0.19.0:

- %KWr.m.0.19.0 = 1: fallback to 0
- %KWr.m.0.19.0 = 0: maintain state

(r = rack address, m = module address)

Operation:
When the AS-i channel changes to STOP:

- With **fallback to 0** option: outputs are forced to 0, then communication stops on the medium.
- With **maintain state** option: outputs are maintained, then communication stops on the medium.

Automatic Slave Addressing

When this function is enabled in the module configuration, it allows the replacement of a faulty slave with a slave of the same type without stopping the AS-i bus, and without the need for special action:

- If the replacement slave is programmed with the same address and has the same profile, it will be automatically inserted in the list of slaves detected, then activated. If this is not the case, the ERR and AS-i LEDs will flash simultaneously.
- If the new slave has never been used (address 0, new slave) and has the same profile, the slave automatically takes on the address of the replaced slave and is therefore in the list of detected slaves and the list of active slaves. If this is not the case, the ERR and AS-i LEDs will flash simultaneously.

These actions are only possible if **one and only one** slave is faulty in the configuration.

Processor Fault

If there is a communication break with the processor, the module switches to OFFLINE mode.
Communication breaks are caused by the following:

- Triggering of the processor watchdog if the **TSX SAY 100** module is positioned in the rack holding the processor.
- Disconnection of the X Bus cable if the **TSX SAY 100** module is positioned in an extension rack.

Module Fault

In the event of a serious fault on the **TSX SAY 100** module (faulty component, etc.) the module stops communicating with the X Bus and with the AS-i Bus. The same type of behavior occurs when a module is removed with the power on.

Removing a Module with Power Switched On

Should a module be removed with the power switched on, communication with the X Bus stops and the processor indicates a module fault.
Communication with the AS-i bus is also interrupted without notice. In this case, the slaves with a watchdog set their outputs in the desired state, and the others remain in position and cannot be set to 0 since the module cannot guarantee communication.

Inserting a module with power switched on

After switching on the **TSX SAY 100** module, it waits to receive configuration from the processor or for one of the push buttons ↑↓ or +/- to be pressed, otherwise it remains at a stop.

AS-i Supply Fault

When an AS-i supply module fault occurs, communication stops with:

- Slaves with a watchdog positioning their outputs in the desired state, except if the slave takes its power from the AS-i.
- Slave commands switch to 0 due to lack of power.

This fault is indicated by the AS-i LED.

Break in the AS-i Medium

There are several ways in which a break in the medium can occur:

1. There is a break in the medium as it exits the module:
   Behavior is the same as that for a power outage, with all slaves disappearing and a supply fault being indicated.
2. There is a break in the medium after the **TSX SAY 100** module and the AS-i supply:
   All slaves disappear and there is no indication of a supply fault.
3. There is a break in the medium after the **TSX SAY 100** module, the AS-i supply and some slaves:
   Slaves located after the break disappear and there is no indication of a supply fault.

3.4 Precautions of use

24 V auxiliary supply

Recommendations

When slaves use a 24 V auxiliary supply, the disappearance of this supply is not managed by the **TSX SAY 100** module.
Information on the disappearance of this supply can be accessed by using a 24 V input.

AS-i V2 bus interface module: TSX SAY 1000

4

At a Glance

Aim of this Chapter

This chapter deals with hardware installation of the TSX SAY 1000 interface module, AS-i V2 bus master on a Premium/Atrium PLC.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
4.1	Description of the TSX SAY 1000 Module	84
4.2	AS-i Bus Diagnostics	95
4.3	Operating Modes of the TSX SAY 1000 Module	97
4.4	Precautions of Use	99
4.5	AS-i V2 Certification	102

4.1 Description of the TSX SAY 1000 Module

At a Glance

Aim of this Section

This Section deals with hardware installation and the characteristics of the TSX SAY 1000 module.

What's in this Section?

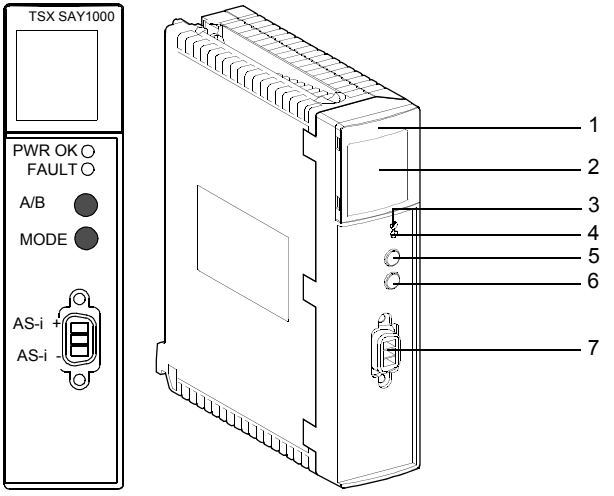
This section contains the following topics:

Topic	Page
Physical description	85
Assembly/installation	87
Connections	88
Displaying Module States	91
Specific Display Panels on the TSX SAY 1000 Module	92
Technical characteristics of As-i V2 bus	93
Personnel Safety	94

Physical description

General

The TSX SAY 1000 module comes as a standard format module.
Illustration:



Front panel

Reference table The following table describes the diagrams above using the numbers shown:

Number	Description
1	A display panel consisting of 4 LEDs for displaying the operating modes of the module: ● RUN LED (green): when lit, this indicates normal functioning of the module. ● ERR LED (red): when lit, this indicates a module error. ● /B LED (green): displays standard or /A address setting slaves. ● I/O LED (red): when lit, this indicates an AS-i bus error.
2	Display panel consisting of 32 LEDs (0 to 31) which enable AS-i bus diagnostics, and display the state of each slave connected to the bus.
3	PWR OK LED (green): when lit, this indicates that power supply is compliant.
4	FAULT LED (red): when lit, this indicates faults associated with the AS-i bus.
5	The A/B push-button can be used to determine the state of devices on the bus by switching from bank A to B.
6	MODE push-button: a prolonged press on this button causes the slaves to be reset and the module to switch to OFF-LINE mode. This then allows the slaves to be programmed via an infra-red interface. The new diagnostics pocket console can also be connected to the bus. To return to normal mode, simply press the button once again for a prolonged period.
7	CANNON SUB D connector for connection to AS-i bus.

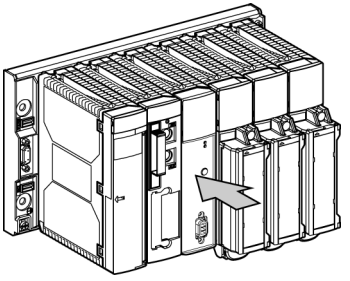
Assembly/installation

General

The module **TSX SAY 1000** can be assembled in any position of a main segment rack of the X bus, except those positions dedicated to the processor and power supply.

	CAUTION
	Forbidden in a remote rack This cannot operate in a remote rack (distance > 100m), and absolutely must be assembled in a main segment rack of the X bus. **Failure to follow this precaution can result in injury or equipment damage.**

The module should be inserted and extracted using a flat or cross tipped screwdriver. These operations can be carried out whether the power is on or off, with no adverse effects on the module or the rack holding it.
Example of mounting a TSX SAY 1000 module:



Note: The module can be mounted and removed with both PLC and AS-i bus power switched on.

Number of Modules per Station

Number of "application-specific" channels supported:

- Premium (See Premium and Atrium using Unity Pro Manual, Processors, racks and power supply modules, Catalog of TSX 57 Processors)
- Atrium (See Premium and Atrium using Unity Pro Manual, Processors, racks and power supply modules, Catalog of Atrium Processors)

Connections

Connecting to the X Bus

The module connects automatically to X BUS once it is inserted into its rack. If the module is inserted into the base rack, connection to the central unit and power supply module is implicit.

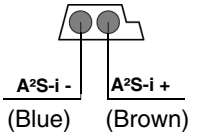
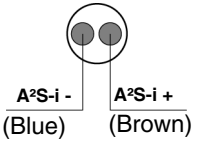
Connection to the AS-i Bus

There is no particular order in which devices (Power Supply, Bus or Slave Master) must be connected to the AS-i network, but the unit as a whole is not guaranteed to be operational during this installation phase.
The AS-i network itself does not need to be grounded. Power supply and PLC devices must, however, adhere to standard installation requirements. You are advised not to place the AS-i network near cables carrying high currents.
The network's connection system is provided for connection to a trapezoid standard multiwire AS-icable (0.75mm to 2.5mm). Regardless of the topology carried over, the combined length of AS-inetwork cabling should not exceed 100m without relay.

AS-i Bus Cables

AS-i bus cables carry the signals and supply the sensors and actuators connected to the bus with 30 VDC.

Types of AS-i cables:

Cable type	Characteristics	Illustration
Polarized AS-i ribbon cable	Color: yellow. Wire cross-section: 1.5 mm^2	
Standard round cable	Wire cross-section: 1.5 mm^2 or 2.5 mm^2	

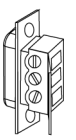
Recommended cable: product reference H05VV-F2x1.5 (flat cable), conforms to the DIN VDE 0281standard. Wire cross-section: 1.5 mm^2.

Cable Routing

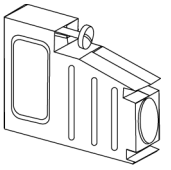
The AS-i cable and the power cables carrying higher power levels must be in separate ducts, which are protected by a metal screen.
When using a shared route for control cables it is essential that the connections on these control links should conform to the technology rules (eg. the discharge diode or limiters on the terminals of self-inductive elements etc.).

Link-up Connector

A set (connector + cover) is delivered with the module, which is used to connect the module to the AS-i bus. This connector must be linked to the cable of the AS-i bus and assembled by the user according to the procedure described later.
Illustration:



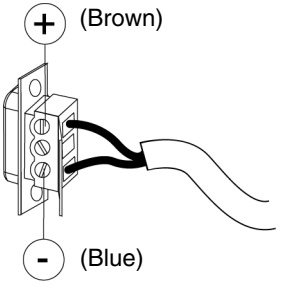
Connector

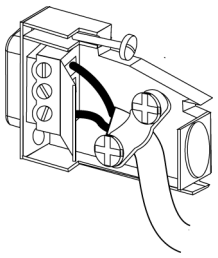
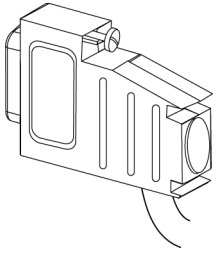
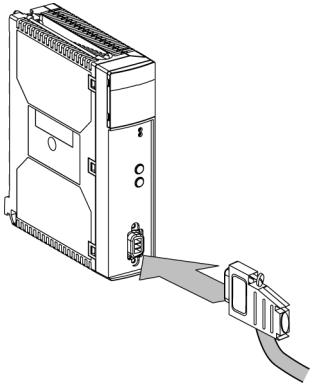


Cover

Connection of Module to Bus

To connect a module to the bus, follow the procedure below:

1	Connect the 2 wires of the AS-i cable to the connector, taking the polarities into account:  In the special event that a shielded cable is used, this should be connected to the central terminal.

2	Mount the connector in its cover and fix the cable to it: 
3	Click the cover shut: 
4	Mount the unit on the module: 

Displaying Module States

Overview

This is carried out with 4 LEDs: RUN, ERR, A/B, I/O which are located on the module. Their state (LED off, flashing or on) provides information on the operating mode of the module.
LED state:

LEDs	**On**	**Flashing(**)**	**Off**
RUN (green)	Module **OK** and configured	Awaiting configuration	Module failure or self-test fault
ERR (red)	Serious non-rectifiable module fault	Rectifiable module fault (Unity Pro configuration, AS-i power supply)	Module **OK**
/B (green)	/B configured extended address settings slaves display	-	/A configured slaves display
I/O (red)	AS-i bus fault	Fault or awaiting user configuration	AS-i Bus **OK**
0 to 31 (green)	Slave number OK (projected, present and active)	Slave Number /OK (*)	Slave number not projected and absent

(**) all LEDs flash during module self-tests.
(*) a slave is declared /OK when one of the conditions required for correct operation is not satisfied, regardless of the error level:

- Error Level 1:
  - Slave declared but not detected,
  - Slave detected but not declared.
- Error Level 2:
  - Slave has different profile from declared slave,
  - Slave profile incompatible with address,
  - Slave with incorrect subprofile.
- Error Level 3:
  - Slave refuses parameters,
  - Slave has auxiliary power supply fault,
  - Analog data fault (channel fault),
  - Other faults linked to analog slave characteristics,
  - Input / Output or auxiliary power supply overload,
  - External error:
    - thermal relay,
    - slave self-tests NOK.

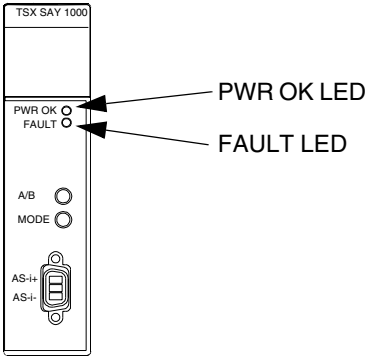
The user must connect his/her programming tool to access detailed information on the faulty slave. The local display can only show a malfunction in its related slave.

Specific Display Panels on the TSX SAY 1000 Module

General

2 LEDs: PWR OK and FAULT give specific information on the TSX SAY 1000 module.

View of the 2 LEDs:



PWR OK LED (green):

LED state	Meaning
LED on	Power Supply consistent
LED off	Power Supply inconsistent

FAULT LED (Red)

LED state	Meaning
LED on	AS-i bus faults
LED off	No faults

Technical characteristics of As-i V2 bus

AS-i V2 Bus Technical Characteristics:

Characteristic	Value
Maximum cycle time of bus: ● - 1 to 19 slaves = 3ms, ● - 20 to 62 slaves = (1+n)*156s where n = number of active slaves.	5 ms for 31 standard or extended address setting slaves, 10 ms for 62 extended address setting slaves.
Maximum number of slaves on the bus:	31 standard address setting slaves, or, 62 extended address setting slaves.
Maximum length of AS-i bus cables: all branches without repeater with two repeaters	 100 meters 300 meters
Maximum number of I/O managed by the bus	standard address setting slaves: 124 inputs +124 outputs extended address setting slaves: 248 inputs + 186 outputs
Nominal bus supply voltage	30 V DC

TSX SAY 1000 Module Technical Characteristics:

Characteristic	Value
Programming the TSX SAY 1000 module	Unity Pro
Response time with 31 slaves (1) for a PLC cycle time of 10 ms	Typically 27 ms, 37 ms maximum
Power consumption 5V PLC	Typically 100 mA / 150 mA maximum
Power consumption on 30 V AS-i/AS-i	Typically 50 mA / 60 mA maximum
Power dissipation	2.5 W max.
Protection against polarity inversion on bus inputs	Yes
Degree of protection	IP20
Isolated voltage	500 V DC
Operating temperature	0 to 60 degrees Celsius
AS-i master profile	M2e
Standards and service conditions	Compliant with those for Premium PLCs (See installation manual for Premium/Atrium PLCs)
(1) Logical response time + time between an AS-i input activated on the bus, processed in the PLC application and applied to an AS-i output.	

Personnel Safety

General

To ensure personnel safety it is essential:

- To earth the ground terminal of the PLC.
- To use an AS-i VLSV (very low safety voltage) supply, nominal voltage 30 VDC.
- For PLCs which are connected to an alternating current network, a differentiel circuit breaker must be placed upstream of this network, and this will cut off the PLC power supply source if ground leakage if detected.
- For PLCs which are connected to a direct current supply source you must ensure that the supply placed upstream of the PLC is VLVS.
- To use certified AS-i products on the bus.

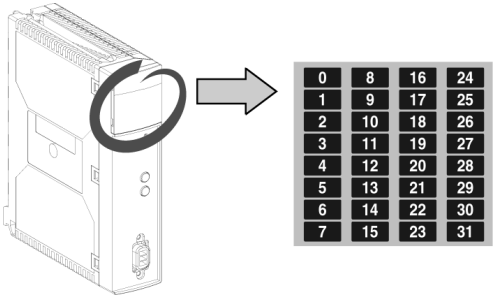
Due to the type of technology and connection, the **TSX SAY 1000** module only receives 5 VDC and its 0V is linked to the PLC ground.

4.2 AS-i Bus Diagnostics

Introduction to AS-i Bus diagnostics

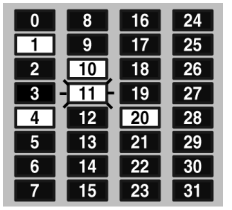
General

The module display panel displays the presence and operating status of each slave on the AS-i bus.
Illustration:



Diagnostics of the AS-i Bus

Table giving diagnostics of module's standard and extended address setting slaves:

View of standard or extended address setting slaves on bank A: /B LED off	**View of extended address setting slaves on bank B: /B LED on**
View of AS-i bus image. Each LED from 1 to 31 corresponds to a standard or extended slave address on the bus. ● LED on: slave active ● LED off: slave not expected and not detected. ● LED flashes rapidly: peripheral fault on slave. ● LED flashing slowly: configuration fault on slave.	View of AS-i bus image. Each LED 1 to 31 corresponds to an extended address setting slave address on the bus. ● LED on: slave present ● LED off: slave not expected and not detected. ● LED flashes rapidly: peripheral fault on slave. ● LED flashing slowly: configuration fault on slave.
Diagnostics example with 5 slaves addressed at 1, 4, 10, 11, 20:  **In brief** : ● LEDs for slaves 1, 4, 10, 20 are lit, therefore these slaves are active, ● slave 11's LED is flashing, so it is faulty, ● the other LEDs are off because no slaves are expected or detected at these addresses.	

4.3 Operating Modes of the TSX SAY 1000 Module

Operating modes of the TSX SAY 1000 module

Output Fallback Position

Since some AS-i V2 slaves have an internal watchdog based on a communication shutdown, when there is a fallback the module will function in the following way:

- fault detection (PLC stop, CPU fault, module fault),
- bus automatically switched OFF LINE by bus master.

Consequences for slaves present on the bus:

- "old generation" slaves: outputs maintained,
- "new generation" slaves: preprogrammed fallback positions are implemented in the slave.

Automatic Slave Addressing

When this function is enabled in the module configuration, it allows the replacement of a faulty slave with a slave of the same type without stopping the AS-i bus, and without the need for special action:

- If the replacement slave is programmed with the same address and has the same profile, it will be automatically inserted in the list of slaves detected, then activated. If this is not the case, the ERR and AS-i LEDs will flash simultaneously.
- If the new slave has never been used (address 0, new slave) and has the same profile, the slave automatically takes on the address of the replaced slave and is therefore in the list of detected slaves and the list of active slaves. If this is not the case, the ERR and AS-i LEDs will flash simultaneously.

These actions are only possible if **one and only one** slave is faulty in the configuration.

Communication Fault

If there is a break in communication with the CPU, following a CPU watchdog (where the **TSX SAY 1000** module is located in the main rack) or a retraction of the X Bus cable (where the **TSX SAY 1000** module is located in the extension rack), the module switches to security mode and stops communication on the AS-i bus.

Module Fault

In the event of a serious fault on the **TSX SAY 1000** module (faulty component, etc.) the module stops communicating with the X Bus and with the AS-i Bus. The same type of behavior occurs when a module is removed with the power on.

Removing a Module with Power Switched On

Should a module be removed with the power switched on, communication with the X Bus stops and the processor indicates a module fault.
Communication with the AS-i bus is also interrupted without notice. In this case, the slaves with a watchdog set their outputs in the desired state, and the others remain in position and cannot be set to 0 since the module cannot guarantee communication.

Inserting a Module with Power Switched On

After voltage is applied to the **TSX SAY 1000** module, it expects to receive a configuration via Unity Pro. If this does not occur it remains in stop mode.

AS-i Supply Fault

When there is an AS-i power supply fault, communication stops and the slaves behave differently.

- slaves with a watchdog position their outputs in the defined state, except if the slave draws its power from the AS-i.
- all slave commands switch to 0 due to lack of power.

From a language point of view, all slaves seem faulty, and the absence of AS-i power supply is indicated in the Channel Status.

Break in the AS-i Medium

There are several ways in which a break in the medium can occur:

1. There is a break in the medium as it exits the module:
   Behavior is the same as that for a power outage, with all slaves disappearing and a supply fault being indicated.
2. There is a break in the medium after the **TSX SAY 1000** module and the AS-i supply:
   All slaves disappear and there is no indication of a supply fault.
3. There is a break in the medium after the **TSX SAY 1000** module, the AS-i supply and some slaves:
   Slaves located after the break disappear and there is no indication of a supply fault.

4.4 Precautions of Use

At a Glance

Aim of this Section

This Section deals with the precautions to be taken when installing an AS-i bus.

What's in this Section?

This section contains the following topics:

Topic	Page
24 VDC Auxiliary Supply	100
Multiple Address Settings	101

24 VDC Auxiliary Supply

Recommendations

When slaves use a 24 VDC auxiliary supply, the disappearance of this supply is not managed by the TSX SAY 1000 module.
Information on the disappearance of this supply can be accessed by using a 24 VDC input.

Multiple Address Settings

Recommendations

When one or more slaves are connected, make sure that you do not assign an address, which is already being used by a slave on the bus.

WARNING

Double slave addressing

The two slaves with identical addresses have the same profile and manage identical I/Os: the AS-i master bus does not detect any error. The two slaves with identical addresses manage different I/Os: the AS-i master bus can detect transmission errors when accessing the I/O from one of the two slaves.

Failure to follow this precaution can result in death, serious injury, or equipment damage.

4.5 AS-i V2 Certification

AS-i V2 Certification

Description

Header:

Vendor:	
Product name:	TSX SAY 1000
Order No. :	
Release:	
Master profile:	
Date:	

List of implemented functions:

No.	**List of implemented functions**	**Mark/ Profile**	**Remark/ implemented by**
A	**Functions or calls at host interface**		
1	Image, Status = Read_IDI ()	*	TSX SAY 1000
2	Status = Write_OD (Image)	*	TSX SAY 1000
3	Status = Set_Permanent_Parameter (S_Addr, S_Param)	*	TSX SAY 1000
4	S_Param, Status = Get_Permanent_Parameter (S_Addr)	*	TSX SAY 1000
5	Status, RS_Param = Write_Parameter (S_Addr, S_Param)	*	TSX SAY 1000
6	Status, S_Param = Read_Parameter (S_Addr)	*	TSX SAY 1000
7	Status = Store_Actual_Parameters	*	TSX SAY 1000
8	Status = Set_Permanent_Configuration (S_Addr,S_ Config)	*	TSX SAY 1000
9	S_Param, Status = Get_Permanent_Parameter (S_Addr)	-	
10	Status = Store_Actual_Configuration ()	-	
11	Status, S_Config = Read_Actual_Configuration (S_Addr)	*	TSX SAY 1000
12	Status = Set_LPS (S_List)	*	TSX SAY 1000
13	Status, S_List = Get_LPS ()	*	TSX SAY 1000
14	Status, S_List = Get_LAS ()	*	TSX SAY 1000
15	Status, S_List = Get_LDS ()	*	TSX SAY 1000
16.0	Status, Flags = Get_Flags ()	*	TSX SAY 1000
16.1	Status, Flag = Get_Flag_Config_OK ()	*	TSX SAY 1000
16.2	Status, Flag = Get_Flag_LDS.0 ()	*	TSX SAY 1000

No.	List of implemented functions	Mark/ Profile	Remark/ implemented by
16.3	Status, Flag = Get_Flag_Auto_Address_Assign ()	*	TSX SAY 1000
16.4	Status, Flag = Get_Flag_Auto_Prog_Available ()	*	TSX SAY 1000
16.5	Status, Flag = Get_Flag_Configuration_Active ()	*	TSX SAY 1000
16.6	Status, Flag = Get_Flag_Normal_Operation_Active ()	*	TSX SAY 1000
16.7	Status, Flag = Get_Flag_APF ()	*	TSX SAY 1000
16.8	Status, Flag = Get_Flag_Offline_Ready ()	*	TSX SAY 1000
16.9	Status, Flag = Get_Flag_Periphery_OK ()	*	TSX SAY 1000
17	Status = Set_Operation_Mode (Mode)	*	TSX SAY 1000
18	Status = Set_Offline_Mode (Mode)	*	TSX SAY 1000
19	Status = Activate_Data_Exchange (Mode)	*	TSX SAY 1000
20	Status = Change_Slave_Address (S_Addr1, S_Addr2)	*	TSX SAY 1000
21.1	Status = Set_Auto_Adress_Enable (Mode)	*	TSX SAY 1000
21.2	Mode = Get_Auto_Adress_Enable ()	*	TSX SAY 1000
22.1	Status, Resp = Cmd_Reset_AS-i_Slave (S_Addr, RESET)	-	
22.2	Status, Resp = Cmd_Read_IO_Configuration (S_Addr, CONF)	-	
22.3	Status, Resp = Cmd_Read_Identification_Code(S_Addr, IDCOD)	-	
22.4	Status, Resp = Cmd_Read_Status (S_Addr, STAT)	-	
22.5	* Status, Resp = Cmd_Read_Reset_Status (S_Addr,STATRES)	-	
22.6	Status, Resp = Cmd_Read_Ext_ID-Code_1 (S_Addr, IDCOD1)	-	
22.7	Status, Resp = Cmd_Read_Ext_ID-Code_2 (S_Addr, IDCOD2)	-	
23	Status, S_List = Get_ List of Periphery Faults ()	*	TSX SAY 1000
24	Status = Write_Extended_ID-Code_1(S_Ext_ID-Code_1)	*	TSX SAY 1000
B	**Integrated support of slave profiles**		
1	Analog slave profile S7.3 support integrated	*	TSX SAY 1000
2	Analog slave profile S7.4 support integrated	-	

(-) = not implemented functions
(*) = implemented functions

AS-i Bus software implementation

III

At a Glance

In This Chapter

This part presents the software implementation of the AS-i bus with Unity Pro software.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
5	Principles of the software implementation of the AS-i Bus	107
6	AS-i bus configuration	117
7	Debugging the AS-i bus	133
8	AS-i operating mode	147
9	AS-i performance	155
10	DFB for AS-i Security Monitor	157
11	Language objects of the AS-i bus	159

Principles of the software implementation of the AS-i Bus

5

At a glance

Aim of this Chapter

This chapter presents the principles of software implementation of the AS-i Bus.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Presentation of the implementation of the AS-i bus	108
Architecture of TSX SAY 100 Module	110
Structure of an AS-i slave	112
Addressing of language objects associated with the slave devices connected to the AS-i bus.	114

Presentation of the implementation of the AS-i bus

Introduction

The AS-i bus (Actuator Sensor-Interface) allows the interconnection on a single cable of sensor devices/actuators at the lowest level of automation. These sensors/actuators will be defined in the documentation as slave devices.
To implement the AS-i application you need to define the physical context of the project into which it will be built (rack, power supply, processor, modules, AS-i slave devices connected to the bus) then ensure its software implementation.
The software implementation of the application modules is performed from the various Unity Pro editors:

- in local mode,
- in online mode.

The following is the recommended order of the implementation phases, but it is possible to modify the order of some phases (for example start with the configuration phase).

AS-i bus implementation principle

The following table shows the different implementation phases of the AS-i bus.

Phase	Description	Mode
Variable Declarations	Declaration of variable of the IODDT type for the project modules and variables.	Local (1)
Programming	Programming the project and functions carried out using the AS-i bus.	Local (1)
Configuring	Declaration of modules and slave devices.	Local
	Configuring the module channels.	
	Entering configuration parameters.	
Association	Association of IODDT's with the configured modules (variables editor).	Local (1)
Generation	Generation (analysis and editing of links) of the project.	Local
Transfer	Transfer of the project to the PLC.	Connected
Adjust and Debug	Debugging the project from debug screens/animation tables.	Connected
	Modification of the program and the adjustment parameters.	
Documentation	Building the documentation file and printing of the various information concerning the project.	Connected (1)
Operation/ Diagnostics	Display of the various information necessary for conducting the project.	Connected
	Project and modules diagnostics.	
Legend		
(1)	These various phases can also be performed in the other mode.	

Architecture of TSX SAY 100 Module

At a glance

The **TSX SAY 100** module operates according to the master/slave mode. The master only controls exchanges on the bus.
The AS-i standard sets several operating levels offered by the master:

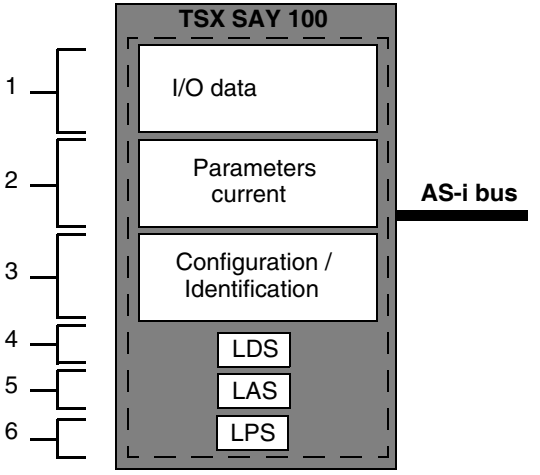
- Profile M0 - Minimum Master: the master only puts forward the configuration of slaves connected to the bus on power-up, and only input/output exchanges.
- Profile M1 - Full Master: this profile covers all the operating functions set by the AS-i standard,
- Profile M2 - Reduced Master: this profile corresponds to profile M0 operating functions with a slave-parametering option.

Note: The TSX SAY 100 corresponds to the profile M2 with the additional option of reading the diagnostics information of the slaves.

The module includes data fields which are used to manage the lists of slaves and I/O data images. This information is stored in a volatile memory.

Illustration of the architecture

The figure below shows the **TSX SAY 100** module architecture.



Description of constituent elements

The table below shows the different elements that make up the architecture of the **TSX SAY 100** module.

Number	Item	Description
1	I/O data	Images of 124 inputs and 124 outputs of AS-i V2 bus.
2	Current parameters	Image of parameters of all slaves.
3	Configuration/ Identification	This field contains all the I/O codes and identification codes for all the detected slaves.
4	LDS	List of all the slaves detected on the bus.
5	LAS	List of active slaves on the bus.
6	LPS	List of slaves provided on the bus and configured by Unity Pro.

Structure of an AS-i slave

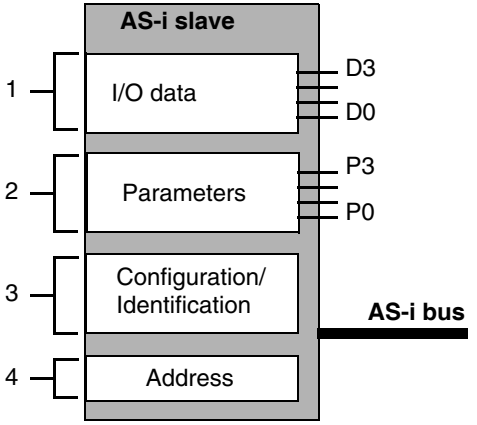
At a glance

The AS-i bus allows interconnection of 31 slave devices each with:
- 4 input bits,
- 4 output bits,
- 4 parametering bits.

So, the AS-i bus is capable of managing a maximum of 248 I/O's.
Each slave has its own address and a profile (defines variables exchange).

Structure illustration

The figure below shows the structure of an AS-i slave.



Description of constituent elements

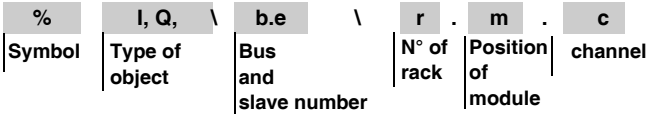
The table below shows the different elements that make up the structure of an AS-i slave.

Number	Item	Description
1	Input/output data	Input data is stored by the slave and made available for the AS-i master. Output data is updated by the master module.
2	Parameters	The parameters are used to control and switch internal operating modes to the sensor or the actuator.
3	Configuration/ Identification	This field contains: ● the code which corresponds to I/O configuration, ● the slave identification (ID) code.
4	Address	Physical address of slave.
Note: The operating parameters, address, configuration and identification data are saved in a non-volatile memory.		

Addressing of language objects associated with the slave devices connected to the AS-i bus.

At a glance

Acquisition of inputs and update of slave device outputs connected to the AS-i bus are carried out automatically. This occurs at the start and end of each cycle respectively of the task in which the **TSX SAY 100** module is configured.
The user program has access to these inputs and outputs via the language objects. Addressing is defined in the following way:



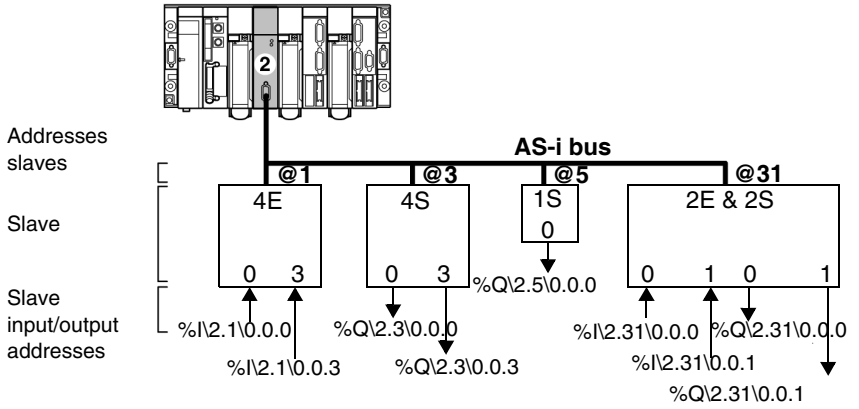
Syntax

The table below describes the different elements that make up addressing.

Family	Item	Values	Indicates
Symbol	%	-	-
Object type	I Q	- -	Picture of the physical input of the module, Picture of the physical output of the module, This information is exchanged automatically each cycle of the task to which they are attached.
Bus and slave No.	b	2 to 999	Bus number.
	e	1 to 31	Slave Number
Rack number	r	0	Virtual rack number.
Module position	m	0	Virtual module position
Channel	c	0 to 3	Input or output number

Example

%I\4.1\0.0.3 indicates: input 3 of bus number 4 of slave 1 and implicitly rack 0 and module 0 of the **TSX SAY 100** module.
%Q\8.31\0.0.0 indicates: output 0 of bus number 8 of slave 31 and implicitly rack 0 and module 0 of the **TSX SAY 100** module.
Illustration:



Note: The physical address of an AS-i slave is programmed by a console.

Multiple Address Settings

When one or more slaves are connected, make sure that you do not assign an address which is already being used by a slave on the bus.
If slave address is doubled then two scenarios may occur :

- the two slaves with identical addresses have the same profile and manage identical I/O's: the AS-i master bus does not detect any error.
- the two slaves with identical addresses manage different I/O's: the AS-i master bus can detect transmission errors when accessing the I/O from one of the two slaves.

AS-i bus configuration

6

At a Glance

Subject of this Chapter

This Chapter describes the Configuration aspect for installing the AS-i bus.

What's in this Chapter?

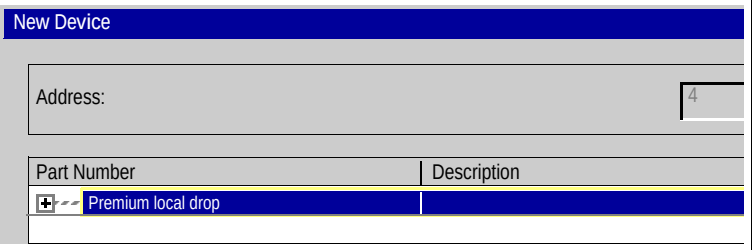
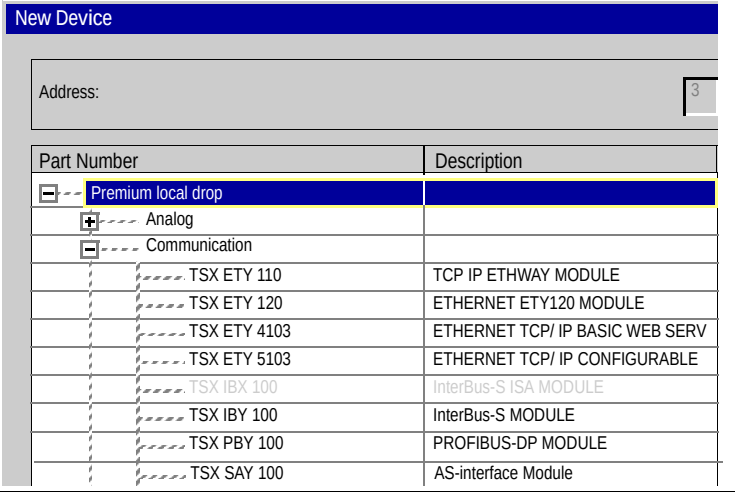
This chapter contains the following topics:

Topic	Page
How to Declare an AS-i Communication Module in the PLC Rack	118
Description of the AS-i communication module configuration screen	119
How to Define a Slave Device on the AS-i Bus	121
AS-i Bus Displayed in the Project Browser	124
How to modify the AS-i Bus software configuration	126
How to access the description of an AS-i slave	127
How to Define a New Slave Profile in a Private AS-i Catalog	129
How to modify AS-i slave general parameters: Automatic addressing	131
How to modify AS-i slave general parameters: Fallback mode	132

How to Declare an AS-i Communication Module in the PLC Rack

Procedure

This operation makes it possible to insert an AS-i communication module in the PLC rack.

Step	Action
1	Open the hardware configuration editor.
2	Select the slot where you wish to insert the module.
3	Select the command **New Device** from the contextual menu. **Result**: the **New Device** window appears. 
4	Open up the line **Premium local input/output station** and the line **Communication** by clicking on the + sign. **Result** : 
5	Select the module **TSX SAY 100** then confirm with the command **OK**.

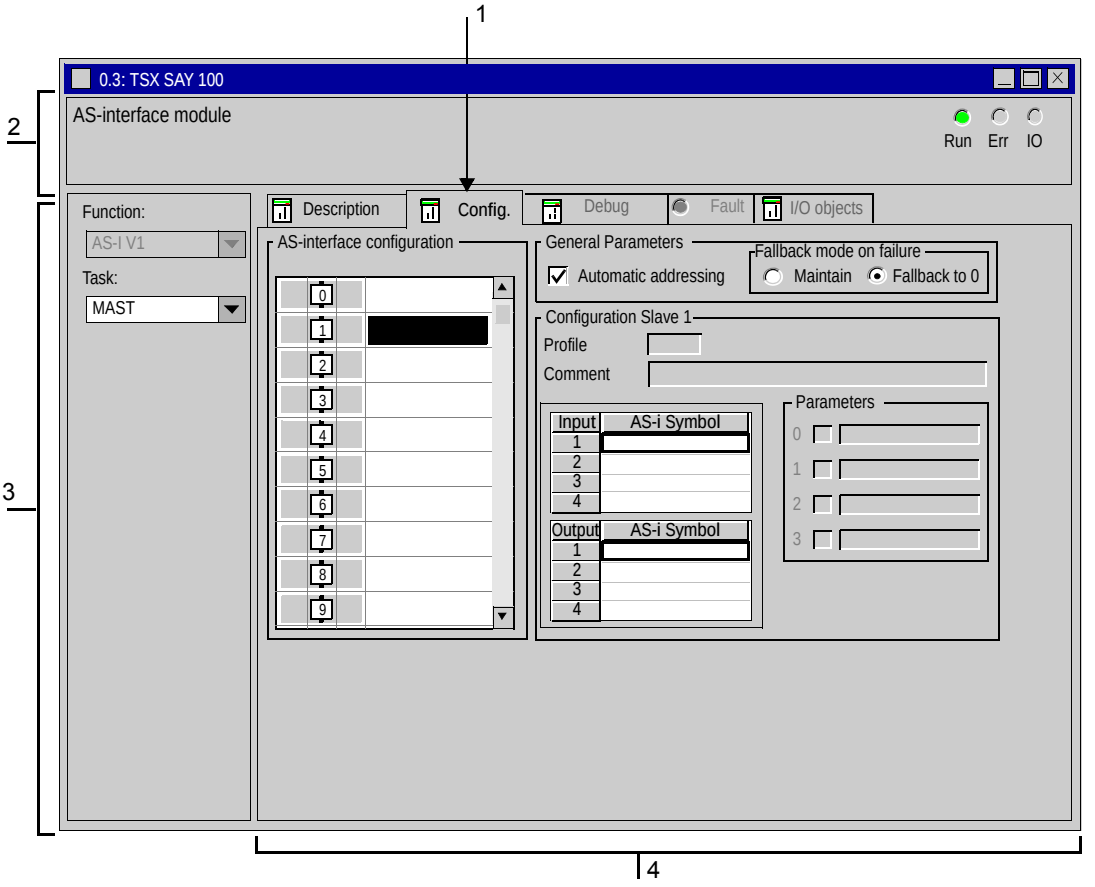
Description of the AS-i communication module configuration screen

At a Glance

The configuration screen is a graphics tool used to configure a module selected in a rack. It gives access to the parameters associated with the module and the slave devices.

Illustration

The figure below represents a configuration screen.



Description

The following table shows the different elements of the configuration screen and their functions.

Address	Item	Function
1	Tabs	The front tab indicates the mode in progress (**Configuration** for this example). Each mode can be selected with the corresponding tab. Possible modes are: • **Configuration**, • **Debug** (or diagnostics), only accessible in online mode, • **Diagnostics** (Default), accessible only in Online mode. The tab **I/O Objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) is used to presymbolize the input/output objects.
2	**module** zone	Specify the abbreviated title of the module and the status of the module in online mode There are three indicators in the same zone that provide the status of module in online: • **RUN** indicates the operating mode of the module. • **ERR** signals an internal fault in the module. • **I/O** signals an external fault in the module or an application fault.
3	**General parameters** zone	makes it possible to select the general parameters associated with the channel : • **Function :** the Bus AS-i function is frozen (grayed). • **Task:** defines the **MAST** or **FAST** task through which the channel's implicit exchange objects will be exchanged.
4	**configuration** field	Makes it possible to configure the channel configuration parameters. Certain choices may be frozen and appear grayed. It is broken down into four fields: • AS-interface configuration, • the selected slave characteristics, • the parameters applied to the chosen slave, • the input and output symbols associated with a device.

How to Define a Slave Device on the AS-i Bus

At a Glance

The Unity Pro software offers a catalog including all of the slaves that are available. This catalogue is structured in families (e.g.: inductive detectors).

The list of slave device families features two specific elements:

- Non-specialized products,
- Private family.

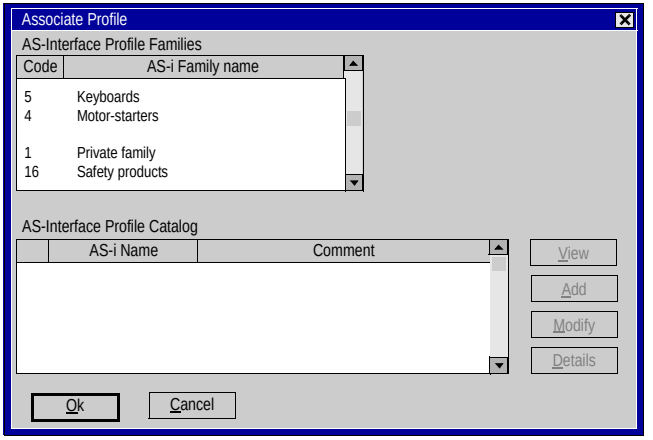
Selecting a non-specialized product enables you to choose an AS-i profile from 240 possibilities.

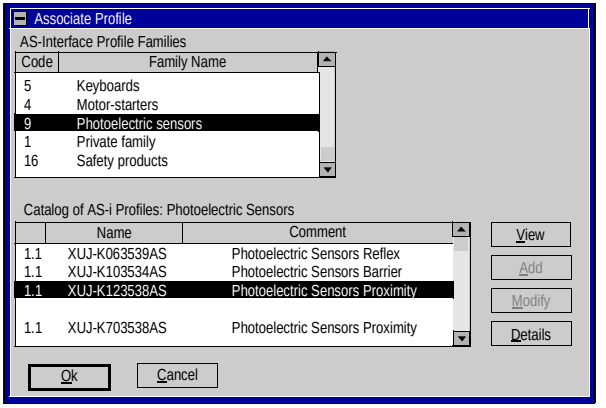
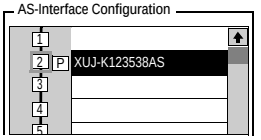
Selecting **Private Family** enables you to manage a specific AS-i device catalogue file via its programming terminal.

Note: A project using the AS-i products from the Private Family catalog is always linked to the usage of this same private family catalog.

Procedure

This operation is used to declare a slave device on the AS-i bus.

Step	Action
1	Open the AS-i module to be configured.
2	Access the configuration screen by clicking the **Configuration** tab.
3	In the **AS-interface configuration** area, double click in the cell corresponding to the new slave's host slot number (1 to 31), or select the cell then click **Edit** → **Add an AS-i Slave**. **Result**: The screen **Associate Profile** appears. 

Step	Action
4	In the **Profile Families** area, select the required family. **Result**: The **Profile Catalog** linked to the selected family appears.  **Note**: It is normal that the following products have been removed since they are no longer manufactured, nor marketed: (Code 7): Signaling and control. (Code 11): Inductive proximity detector. (Code 3): Modular IP67 M12 Interface. (Code 10): IP67 M12 Interface. (Code 15): IP67 M8 Interface. (Code 5): Keyboard. (Code 9): Photo-electric Detector. (Code 6): Only old XVA-S102s have been removed. The products are still listed, but can no longer be accessed via the families, this having been removed. It is no longer possible to add the above products, but if the application was using them, then these products should appear on the configuration screen (no blanks) and everything will operate normally.
5	In the **AS-Interface Profile Catalog** select the required device.
6	Confirm the selection with**OK**. **Result**: The slave device is defined in its slot, the reference of the connected device appears opposite the number of the slave. 
7	To connect other slave devices to the AS-i bus, repeat the procedure from step 3.

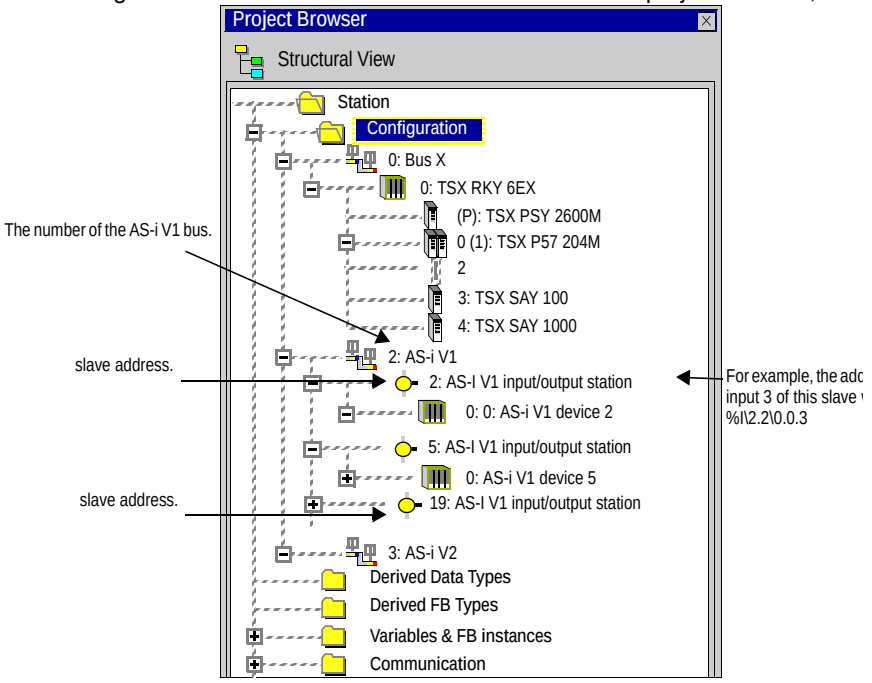
AS-i Bus Displayed in the Project Browser

At a Glance

When you declare a **TSX SAY 100** module on the PLC rack, the AS-i bus is represented in the **configuration** directory of the project browser. The number of the AS-i bus is calculated automatically by Unity Pro. **This value may be modified**.

After having **declared** all the slaves on the AS-i bus and **validated** the configuration, the AS-i slaves also appear on the AS-i bus of the project browser. Each slave appears with its address number. Displaying the AS-i bus and slaves enables you to know at a glance their topological addressing.

The following illustration shows the AS-i bus and slaves in the project browser;



In the project browser, the AS-i bus is represented by **AS-i V1** and the AS-i V2 is represented by **AS-i V2**.

Modifying the Bus Number

It is possible to modify the AS-I bus number by following the below instructions:

Step	Action
1	Close the AS-I module configuration screen (if it is open).
2	In the project browser, right click on the AS-I module and select properties.
3	In the **Bus properties** window, select the desired bus number (between 1 and 999).
4	Click on **OK** to confirm the modification. **Result:** the addresses of the slave devices directly associated with the AS-I bus module are automatically modified.

Note: You are advised not to use the number 2 bus for the AS-I bus because this number is reserved for the Fipio bus.

How to modify the AS-i Bus software configuration

Introduction

The Unity Pro software offers, from the module's configuration screen, a group of functions which allow you to easily modify, in local mode, the software configuration of the AS-i bus.

Note: Standard Windows keyboard shortcut keys (Del, Ctrl-X, Ctrl-C, Ctrl-V) are also available for the following operations:

Procedure for deleting a slave

This operation allows the deletion of a slave declared on an AS-i Bus.

Step	Action
1	Select the slave to be deleted.
2	Select the command **Edit** → **Delete an AS-i slave**.

Procedure for moving a slave

This operation allows the movement of a slave declared on an AS-i Bus.

Step	Action
1	Select the slave to be moved.
2	Select the command **Edit** → **Cut an AS-i slave**.
3	Select the new required slot.
4	Select the command **Edit** → **Paste an AS-i slave**.

Procedure for copying a slave

This operation allows the copying of a slave declared on an AS-i Bus.

Step	Action
1	Select the slave to be copied.
2	Select the command **Edit** → **Copy an AS-i slave**.
3	Select the slot of the new slave.
4	Select the command **Edit** → **Paste an AS-i slave**.

How to access the description of an AS-i slave

At a Glance

The Unity Pro software allows access to all the information relating to an AS-i device such as:

- the definition of a profile,
- the details of a profile.

Definition of a profile

A profile is defined by:

- its name,
- a comment (optional),
- identifiers (IO, ID),
- a number of inputs and/or outputs,
- operation parameters.

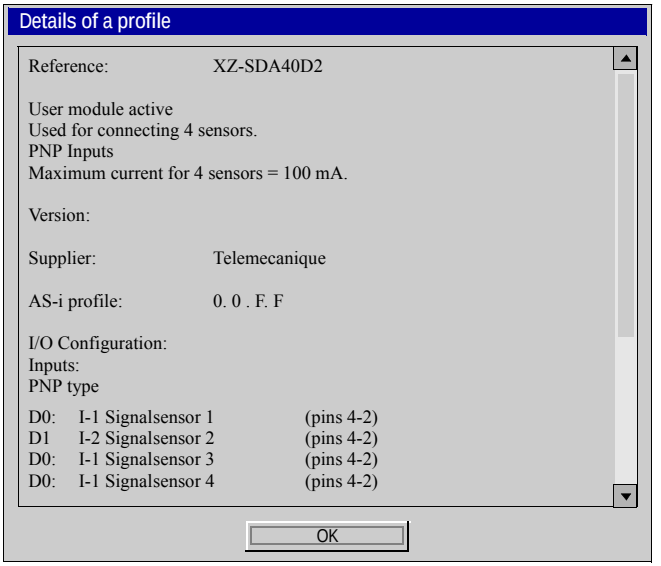
Procedure for accessing information on a profile

The following table shows the procedure for displaying the characteristics of a slave device.

Step	Action
1	Open the AS-i module to be configured.
2	Access the configuration screen by clicking on the tab **Configuration**.
3	Double click on the required slave. **Result:** The window **Associate a profile** displays and highlights the device in question.
4	Click on the button: • **View** to access definition information, • **Details** to access all the information.

Details of a profile

The **Details** function allows access, for a given slave, to all the information presented in the catalog file.



How to Define a New Slave Profile in a Private AS-i Catalog

At a Glance

The Unity Pro software offers the possibility of defining a slave profile that is not provided in the standard catalog.

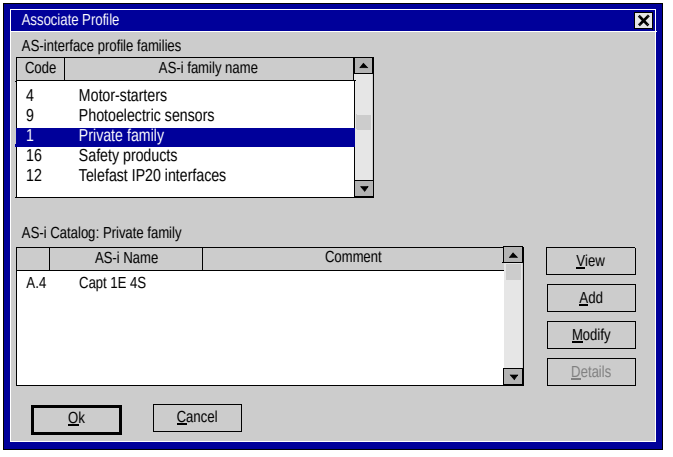
The new profile thus defined is added to the catalog in **Private family**.

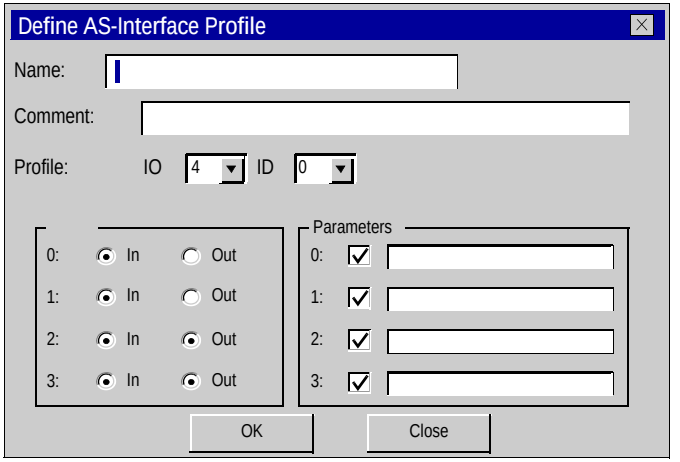
This profile can then be used as a standard catalog profile.

Note: A profile cannot be deleted, only their name and comment can be changed. A profile created and added in the catalog is closely linked to the PC on which it was created, and to the programming software. A project converted using Unity Pro or opened on another PC that does not have the same catalog will not display the profile's name and comment.

Procedure

The following table presents the procedure for defining a slave profile, which is not provided in the standard catalog.

Step	Action
1	Access the AS-i module's hardware configuration screen.
2	Double click in a slave's host cell (1 to 31). **Result**: the **Associate Profile** screen appears.
3	Select **Private family** in the **AS-i family name** field. **Result**: the **AS-i catalog** linked to the selected family appears. 

Step	Action
4	Click on the button **Add**. **Result**: the **Define AS-Interface Profile** window appears. 
5	Enter: ● The name of the new profile, ● A comment (optional).
6	Select: ● The **IO** code (corresponds to the input/output configuration), ● The **ID** code (identifier).
7	For each parameter, define: ● The system's acknowledgement (box checked), ● A label (optional).
8	Confirm the introduction of a new profile by clicking **OK**.

How to modify AS-i slave general parameters: Automatic addressing

At a Glance

Each slave on the AS-i bus must be assigned (via configuration) a unique physical address. This must be the same as the one declared in Unity Pro.

Unity Pro software offers an automatic slave addressing utility so that an AS-i console does not have to be used.

The automatic addressing utility, which can be accessed via Unity Pro:

- replacing a faulty slave (See *Automatic replacement of a faulty AS-i slave, p. 143*),
- inserting a new slave (See *How to insert a slave device into an existing AS- i configuration , p. 144*).

Note: A new configuration with automatic addressing will not be accepted if one or more slaves with a 0 address are on the bus. In this case, the `Configuration refused by module` message appears.

Procedure

The table below shows the procedure for setting the **Automatic addressing** parameter.

Step	Action
1	Open the AS-i module to be configured.
2	Access the configuration screen by clicking on the tab **Configuration**.
3	Click on the **Automatic addressing** check box found in the **General parameters** zone. **Result**: The **Automatic addressing** utility will be activated (box checked) or disabled (box not checked. **Note**: By default, the **Automatic addressing** parameter has been selected in the configuration screen.

How to modify AS-i slave general parameters: Fallback mode

At a Glance

This parameter sets the fallback mode which slave outputs take on while switching to STOP mode or if there is a PLC fault.

Possible modes are:

- **Fallback to 0**: AS-i slave outputs which are present on the bus are set to 0 (%Q objects are not modified),
- **Maintain state**: AS-i slave outputs remain in the state they were before STOP mode was engaged.

Note: The fallback mode for slaves which do not have a watchdog (AS-i bus monitoring function) is not guaranteed in case of an AS-i bus fault, or AS-i power supply failure. For slaves with a watchdog, the fallback position is preset within the device.

Procedure

The table below shows the procedure for setting the **Fallback mode** assigned to slave device outputs.

Step	Action
1	Access the AS-i communication module's configuration screen.
2	Click on the **Fallback to 0** / **Maintain state** check-box found in the **General parameters** zone, in the **Fallback mode on failure** field. **Result**: The fallback mode selected will therefore be assigned to the slave device.

Debugging the AS-i bus

7

At a Glance

Subject of this Chapter

This Chapter describes the Debug aspect of the AS-i bus.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Introduction to the Debug function	134
Description of the debug screen of an AS-i module	135
How to access the functions of the module diagnostics and channel diagnostics of an AS-i device	137
Displaying the status of the slaves	139
How to Access Adjustment of an AS-i Device's Parameters	140
How to Access the AS-i Channels' Forcing/Unforcing Function	141
How to Access the SET and RESET Commands of the AS-i Channels	142
Automatic replacement of a faulty AS-i slave	143
How to insert a slave device into an existing AS- i configuration	144
How to modify the address of an AS-i device	145

Introduction to the Debug function

At a Glance

The **Debug** function is used for each AS-i communication module present in the project:

- to display the slave state (connection, parameters etc.),
- to access the adjustment function for the selected channel (channel forcing etc.).

The function also accesses module diagnostics in the event of a fault.

Note: This function is only available in online mode.

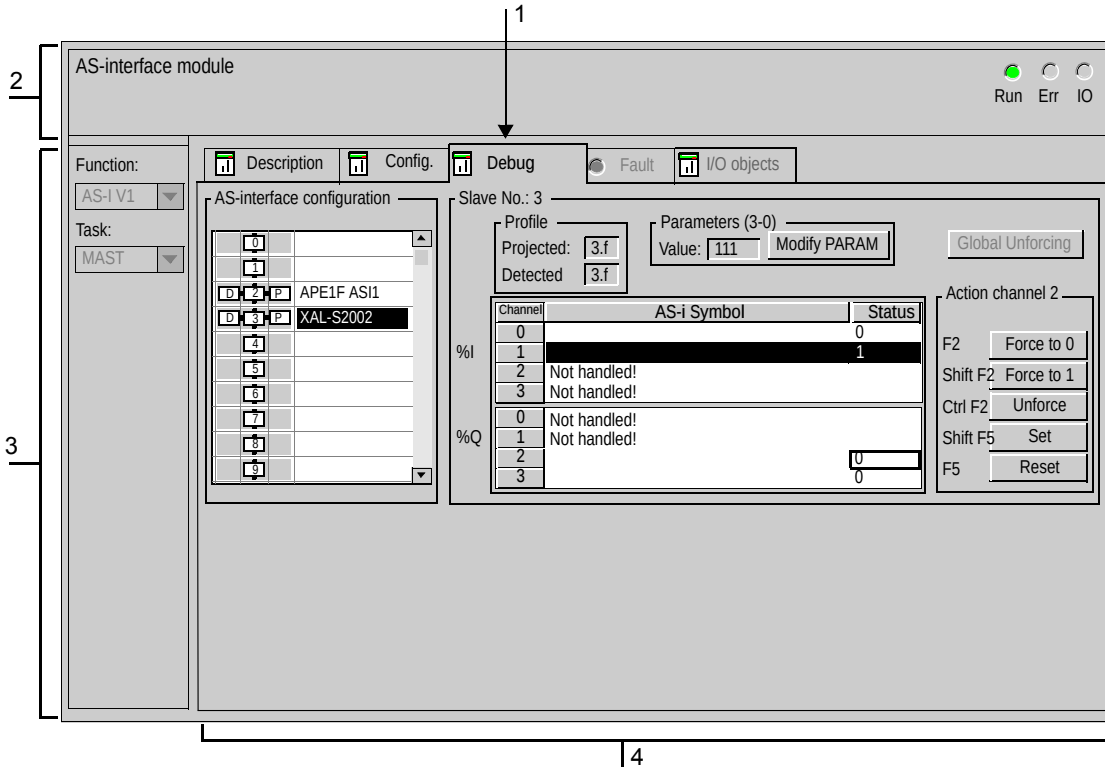
Description of the debug screen of an AS-i module

At a Glance

The debug screen dynamically displays the status of the AS-i module and the devices which are connected to the bus.
It also allows access to the adjustment of the slave parameters and to the channel commands (forcing the input or output value, Set/Reset of an output,).

Illustration

The figure below shows an example of a debug screen.



Description

The following table shows the different elements of the debug screen and their functions.

Address	Item	Function
1	Tabs	The front tab indicates the mode in progress (**Debug** for this example). Each mode can be selected with the corresponding tab. Possible modes are: ● **Debug** only accessible only in online mode. ● **Diagnostics** (Default), accessible only in Online mode, ● **Configuration**. The tab **I/O Objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) is used to presymbolize the input/output objects.
2	**module** zone	Specify the abbreviated title of the module. In the same field there are 3 LEDs which provide information on the operating mode of the module ● **RUN** indicates the operating mode of the module, ● **ERR** signals an internal fault in the module, ● **I/O** signals an external fault in the module or an application fault.
3	**General parameters** zone	Specify the parameterization of the task **MAST** or **FAST** configured for the AS-i communication channel.
4	**AS-i configuration and slave** area	Displays the slave devices connected to the bus. Displays the status of the slave's channels and gives access to the debug functions.

How to access the functions of the module diagnostics and channel diagnostics of an AS-i device

At a Glance

The functions of the module or channel diagnostics display the current errors in which are classed according to their category:

- internal errors (internal software errors, communication error with the processor, configuration, parametering or command error),
- external faults (slave device failed, AS-i power supply switched off, terminal error, difference between physical configuration and Unity Pro configuration),
- other errors (module absent or switched off).

A faulty module and channel become apparent when certain lights change to red, such as:

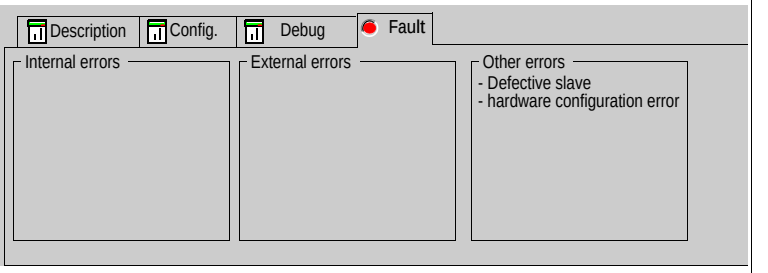
- in the rack configuration screen by the presence of a red square in the position of the faulty AS-i module,
- in all the module level screens (**Description** and **Default** tabs),
  - in the module zone with the LED **I/O**.
- in all the channel level screens (**Description**, **Config**, **Debug** and **Default** tabs),
  - in the module zone with the **I/O** LED,
  - in the channel zone with the LED of channel fault.
- in the faulty screen accessible with the **Fault** tab where the fault diagnostics are described.

The fault is also signaled:

- on the module, through the centralized display,
- by the dedicated language objects: **CH_ERROR** (%Ir.m.c.ERR) and module error **MOD_ERROR** (%Ir.m.MOD.ERR), **%MWr.m.MOD.2**, etc., and the status words (See *Details of the Language Objects of the IODDT of type T_GEN_MOD, p. 187*).

Procedure for accessing the diagnostics

The following table presents the procedure to access the screen **Diagnostics** of the AS-i channel and module.

Step	Action
1	Open the AS-i module to be diagnosed.
2	Access the configuration screen by clicking on the **Fault** tab . **Result**: The list of module errors appears. 

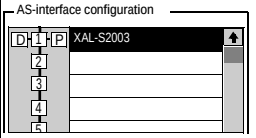
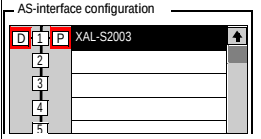
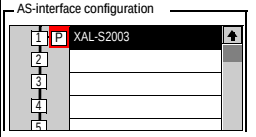
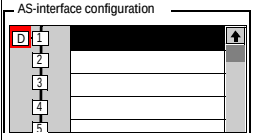
Displaying the status of the slaves

At a Glance

The lower part of the communication module's debugging screen is reserved for the diagnostics of the AS-i bus.
The slave devices connected to the bus are displayed in the **AS-interface configuration** field. On each side of the slave number, two different icons are displayed which show that the slave was specified or detected.

Displaying slave status

For each slave device, one of the following four cases can occur:

Case	Illustration	Explanation
1	Slave status: 	The slave **P** specified in configuration and the detected slave **D** are identical.
2	Slave status: 	The slave **P** specified in configuration and the detected slave **D** are not identical. The slave is declared faulty (1).
3	Slave status: 	A slave **P** is specified at configuration but no slave is detected. The slave is declared faulty (1).
4	Slave status: 	An additional slave, not specified at configuration, is connected to the bus. The slave is declared faulty (1).
Key:		
(1)	When a slave is faulty, the icons situated beside the number turn to red.	
Note: The **Profile** field in the **Slave zone** of the debugging screen allows you to check if the profiles of the specified (**Projected**) slave and the **Detected** slave are really identical.		

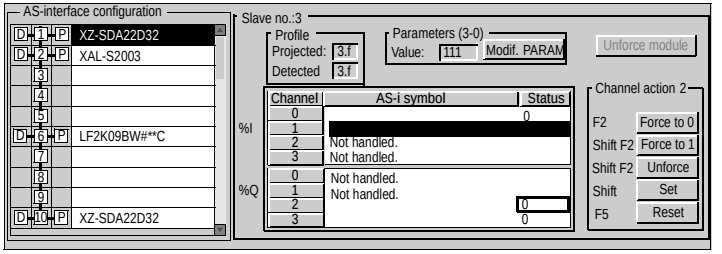
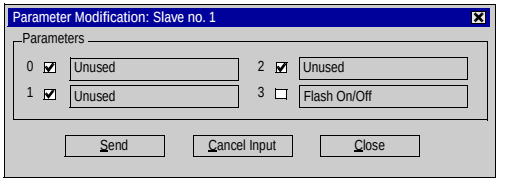
How to Access Adjustment of an AS-i Device's Parameters

At a Glance

The debug screen of an AS-i module allows, amongst other things, access to modification of a slave's parameters.

Procedure

The following table shows the procedure for modifying the parameters of a slave device declared faulty.

Step	Action
1	Open the AS-i module to be parameterized.
2	Access the diagnostics screen by clicking on the**Debug** tab .
3	Select the faulty slave in the **AS-interface configuration** area. **Result**: in the slave zone of the debug screen, it is possible to read all the information relating to the selected slave. 
4	Click on **Modif. PARAM** located in the **Parameters** field of the slave zone. **Result**: the **Modify Parameters** window appears. 
5	Modify the required parameters.
6	Click on **Send** to recognize the new values.

How to Access the AS-i Channels' Forcing/Unforcing Function

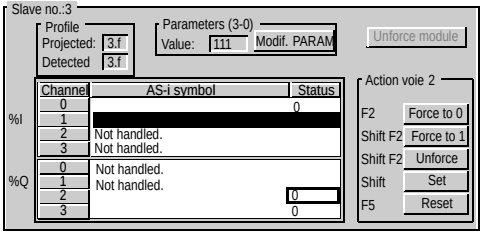
At a Glance

This function supports the modification of the state of the channels, which are linked to a AS-i slave.
The different commands, which are available are:

- For a channel:
  - Forcing to 0,
  - Forcing to 1,
  - Unforcing.
- For all the channels (when at least one channel is forced):
  - Unforce module.

Procedure

The following table shows the procedure to follow to force or unforce the channels linked to an AS-i slave.

Step	Action for one channel	Action for a group of channels
1	Open the AS-i module to be parameterized.	
2	Access the diagnostics screen by clicking on the**Debug** tab.	
3	Select a slave in the **AS-interface configuration** area.	
4	Select the channel to modify in the slave zone table. **Result**: the channel can be modified using the buttons in the **Channel action** field. 	Click on the button **Global unforcing** found in the slave zone.
5	Select the required function (buttons **Force to 0** or **Force to 1** in the **Channel Action** field.	

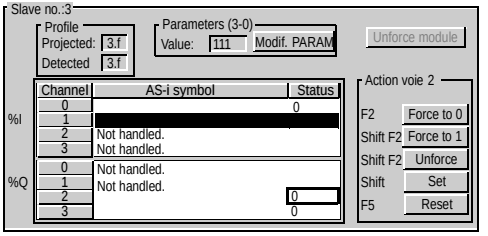
How to Access the SET and RESET Commands of the AS-i Channels

At a Glance

These commands allow the assignment of the values 0 (RESET) or 1 (SET) to the channels of an AS-i slave.
The status of the output affected by one of these commands is temporary and can be modified at any time by the project.

Procedure

The following table presents the procedure for assigning a value of 0 or 1 to the selected AS-i slave's channels.

Step	Action
1	Open the AS-i module to be debugged.
2	Access the debug screen by clicking on the**Debug** tab.
3	Select a slave in the **AS-interface configuration** area.
4	Select the channel to be modified in the **Slave** zone table. **Result**: the channel can be modified using the buttons in the **Channel action** field. 
5	Select the required function (**Set** or **Reset** buttons) in the **Channel action** field.

Automatic replacement of a faulty AS-i slave

Principle

When a slave has been declared faulty, it can be automatically replaced with a slave of the same type.
This happens without the AS-i bus having to stop and without any particular manipulation as long as the **Automatic addressing** (See *How to modify AS-i slave general parameters: Automatic addressing, p. 131*) utility of the configuration mode is active.
Two options are available:

- The replacement slave is programmed with the same address using the pocket programmer, and has the same profile as the faulty slave. It will then be automatically inserted into the list of detected slaves (LDS) and list of active slaves (LAS).
- The replacement slave is blank (address 0, new slave) and has the same profile as the faulty slave. It will automatically assume the address of the replaced slave, and will then be inserted into the list of detected slaves (LDS) and the list of active slaves (LAS).

How to insert a slave device into an existing AS- i configuration

At a Glance

It is possible to insert a device into an existing AS-i configuration without having to use the pocket programmer.
This operation is possible once:

- The configuration mode's **Automatic addressing** (See *How to modify AS-i slave general parameters: Automatic addressing, p. 131*) utility is active
- A single slave is absent in the physical configuration
- The slave which is to be inserted is specified in the Unity Pro configuration
- The slave has the profile expected by the configuration
- The slave has the address 0

The AS-i module will therefore automatically assign to the slave the value predefined in the configuration.

Procedure

The following table shows the procedure for making the automatic insertion of a new slave effective.

Step	Action
1	Add the new slave in the configuration screen in offline mode.
2	Carry out a configuration transfer to the PLC in online mode.
3	Physically link the new slave with address 0 to the AS-i bus.

Note: A project can be modified by carrying out the above manipulation as many times as necessary.

How to modify the address of an AS-i device

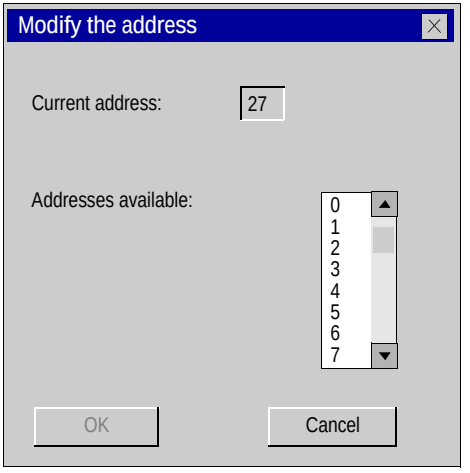
At a Glance

This command allows the movement of the selected AS-i device to another available address. This address modification can only be done online and not in configuration: thus physical configuration and software configuration are different.

Note: This function is only operational for the **TSX SAY 100** modules, in a version higher than 02.

Procedure

The following table shows the procedure for modifying the address of an AS-i slave device.

Step	Action
1	Access the debug screen of the TSX SAY 100 module.
2	Select a slave in the zone **AS-interface configuration** then execute the command **Edit** → **Modify the AS-i slave address**. **Result**: The selection screen for a new address is displayed. 
3	Select the required address in the list of **Available addresses** (use the scroll bar if necessary).
4	Confirm the selection with **OK**.

AS-i operating mode

8

At a glance

Aim of this Chapter

This chapter introduces the different AS-i function operating modes.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
AS-i Operating Mode: General	148
AS-i protected Mode	150
AS-i tes mode of wiring	151
AS-i Offline Operating Mode	152
AS-i Data Exchange Off Operating Mode	153

AS-i Operating Mode: General

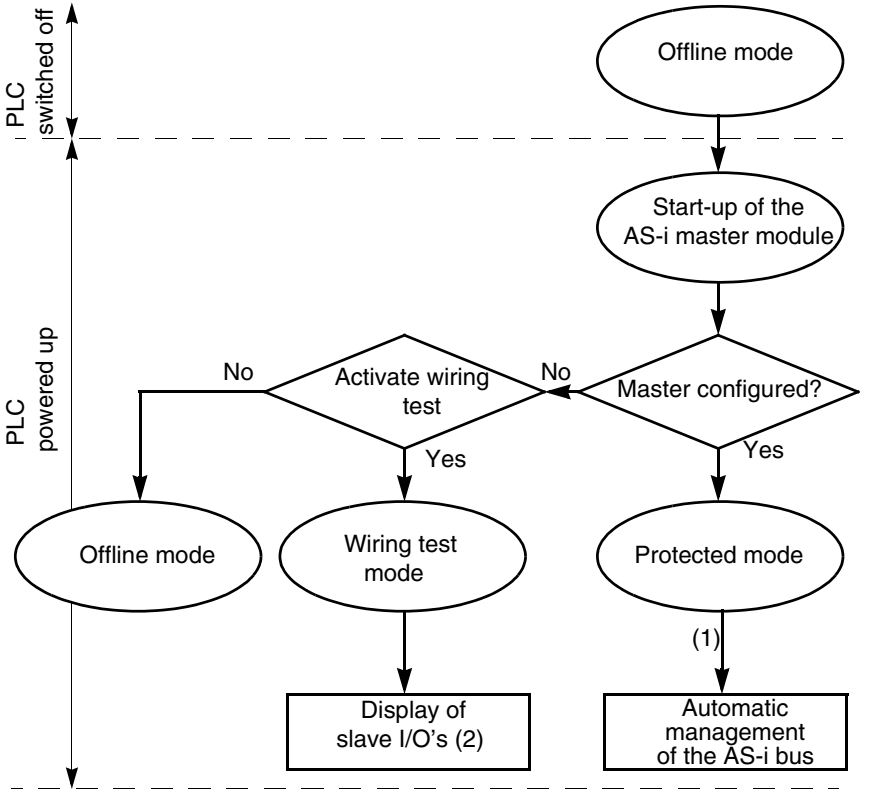
At a Glance

The AS-i function allows four operating modes, each one fulfilling particular needs. These modes are:

- protected mode,
- wiring test mode (accessible using a button positioned on the front of the module),
- offline mode,
- data exchange off mode.

Mode of Functioning

The figure below shows the general operating diagram of the AS-i bus.



(1) : It is possible to toggle between the protected mode and the offline (See *AS-i Offline Operating Mode, p. 152*) or data exchange off AS-i (See *AS-i Data Exchange Off Operating Mode, p. 153*) modes.

(2) : The module leaves the configuration mode and goes into protected mode if it receives a configuration.

Correspondence between PLC operating modes and the AS-i bus

The table below shows the correspondence between Premium or Atrium PLC operating modes and those of the AS-i V2 bus.

PLC	**AS-i Bus**
configured mode (1)	Protected mode
non configured mode (1)	Wiring test mode (Configuration mode)
Legend:	
(1)	These PLC notions (configured, non-configured) concern the declaration of the module and slave devices in the Unity Pro project hardware configuration screen.

AS-i protected Mode

At a glance

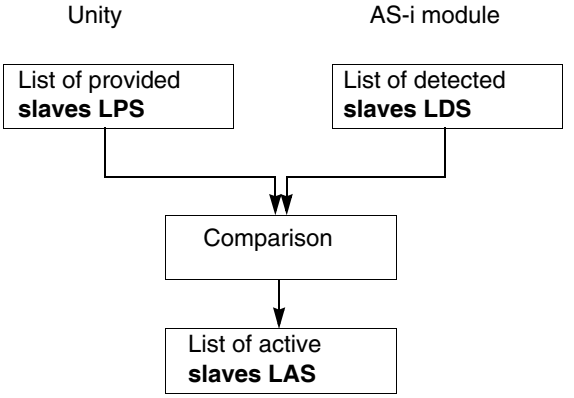
The AS-i protected operating mode is the mode generally used for a project which is running.
It assumes that the AS-i module is configured in Unity Pro.
This:

- continually checks that the list of detected slaves is the same as the list of provided slaves,
- monitors the power supply.

In this mode, a slave will only be activated if it has been declared in the configuration and been detected.

Principle of activating a slave

The diagram below shows the principle of activating AS-i slaves.



AS-i test de wiring mode

At a glance

The **test mode of wiring** , which is particularly useful when starting up new installations, allows the display on the front of the TSX SAY 100 module:

- the slaves provided and detected,
- the slaves not provided and not detected,
- the slaves provided and not detected, or not provided and detected.

For each slave present on the bus, this mode also allows display of the status of the input/output bits.

Note: Lists of LDS and LAS and the adjustment parameters of the slaves are not accessible in this mode.

Access conditions:

The **test mode of wiring** is an AS-i function that is accessible when:

- the **TSX SAY 100** module is in **no configured** mode.
  This mode is obtained when:
  - the **TSX SAY 100** module has not been declared in the project,
  - the PLC processor is missing.
- each device has a separate address.
- address 0 is not used.

AS-i Offline Operating Mode

At a glance

The offline AS-i bus mode is an advanced operating mode which can be used in debugging or maintenance.

Note: The user must be thoroughly familiar with AS-i communication.

Principle

When offline mode is engaged, the module first resets all the present slaves to zero and stops exchanges on the bus.
When in offline mode, the picture of the I/O's in the module is frozen in the state at the time of entering this mode.
When exiting from the module, if the list of present slaves (LPS) is equal to the list of detected slaves (LDS), the system restarts. Otherwise a fault is generated and the diagnostics or configuration mode must be referred to.

Procedure

The offline operating mode is accessible either:
- from the Unity Pro application software by acting on bits 0 and 1 of the %MWr.m.0.24 word,
- or automatically on detection of an AS-i supply fault.

Reminder: Bit 7 of word %MWr.m.0.3 set on 1 indicates that the AS-i bus is in **offline** mode.

AS-i Data Exchange Off Operating Mode

At a glance

The data exchange off AS-i mode is an advanced operating mode which can be used in debugging or maintenance.

Note: The user must be thoroughly familiar with AS-i communication.

Principle

When the data exchange off mode is engaged, exchanges on the bus continue to function, but data is no longer refreshed.

Procedure

The data exchange off operating mode can be accessed from Unity Pro application software by acting on bits 2 and 3 of the word %MWr.m.0.24.

Reminder: Bit 8 of word %MWr.m.0.3 set on 1 indicates that the AS-i bus is in **data exchange off** mode.

AS-i performance

9

Performances of the AS-i bus

Introduction

The AS-i bus is independently managed by the master. This exchanges data on each cycle with each slave device configured on the bus (in ascending order of slave address number).

AS-i Scanning Time

Scanning time **t** represents the exchange time between the master and **n** slaves (31 maximum).
So, for:

- t = 156 microseconds x (n+2), if n < 31,
- t = 156 microseconds x (n+1), if n = 31.

So the scanning time cannot exceed 5 ms.

AS-i Response Time

The response time **T** represents the AS-i cycle time.
This includes:

- the bus scanning time,
- the update of the AS-i module internal memory,
- the PLC cycle.

Example

The table below shows three examples of T response time for a PLC task lasting 10 ms, 30 ms, and 60 ms.
This T time is for a bus loaded with 31 slaves operating normally with no link faults.

PLC task	Typical response time	Maximum response time
10 ms	35 ms	56 ms
30 ms	65 ms	96 ms
60 ms	110 ms	156 ms

DFB for AS-i Security Monitor

10

AS-i security monitor DFBs

At a Glance

The DFBs for the AS-i bus security monitor are described in the section on AS-i V2 (See *DFB for AS-i Security Monitor, p. 247*).

Language objects of the AS-i bus

11

At a glance

Aim of this Chapter

This chapter describes the language objects associated to the AS-i bus as well as the different ways of using them.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
11.1	Language objects and IODDT of the AS-i communication	160
11.2	The language objects and generic IODDT applicable to all communication protocols	169
11.3	Language objects and IODDT's associated to the AS-i bus.	173
11.4	The IODDT type T_GEN_MOD applicable to all modules	187

11.1 Language objects and IODDT of the AS-i communication

At a glance

Aim of this Section

This sub-section presents general information on the language objects and IODDT associated to the AS-i bus.

What's in this Section?

This section contains the following topics:

Topic	Page
Presentation of language objects of the AS-i bus	161
Implicit exchange language objects associated with the application-specific function	162
Explicit exchange language objects associated with the application-specific function	163
Management of Exchanges and Reports with Explicit Objects	165

Presentation of language objects of the AS-i bus

General

The IODDT's are predefined by the constructor and contains language objects of inputs/outputs belonging to the channel of an application module. AS-i communication AS-i has two associated IODDT's :

- `T_COM_STS_GEN` which applies to all the communications protocols,
- `T_COM_ASI_V1` specific to AS-i communication and the TSX SAY 100 module.

Note: IODDT variables can be created in two different ways:
- Using the **I/O objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) tab,
- Data Editor (See Unity Pro, Operating Modes Manual, Creation of an IODDT type data instance).

Language object types

Each IODDT contains a set of language objects allowing its operation to be controlled and checked.

There are two types of language objects:

- **implicit exchange objects**, which are automatically exchanged at each cycle of the task associated with the module,
- **explicit exchange objects**, which are exchanged when requested to do so by the project, using explicit exchange instructions.

Implicit exchanges concern the states of the modules, communication signals, slaves, and so on.

Explicit exchanges allow module parametering and diagnostics.

Implicit exchange language objects associated with the application-specific function

At a Glance

An integrated application-specific interface or the addition of a module automatically enhances the language objects application used to program this interface or module.

These objects correspond to the input/output images and software data of the module or integrated application-specific interface.

Reminders

The module inputs (%I and %IW) are updated in the PLC memory at the start of the task, the PLC being in RUN or STOP mode.

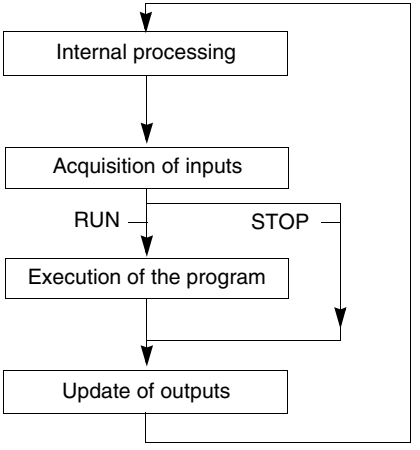
The outputs (%Q and %QW) are updated at the end of the task, only when the PLC is in RUN mode.

Note: When the task occurs in STOP mode, either of the following are possible, depending on the configuration selected:

- Outputs are set to fallback position (fallback mode),
- Outputs are maintained at their last value (maintain mode).

Figure

The following diagram shows the operating cycle of a PLC task (cyclical execution).



Explicit exchange language objects associated with the application-specific function

At a Glance

Explicit exchanges are exchanges performed at the user program's request, and using instructions:

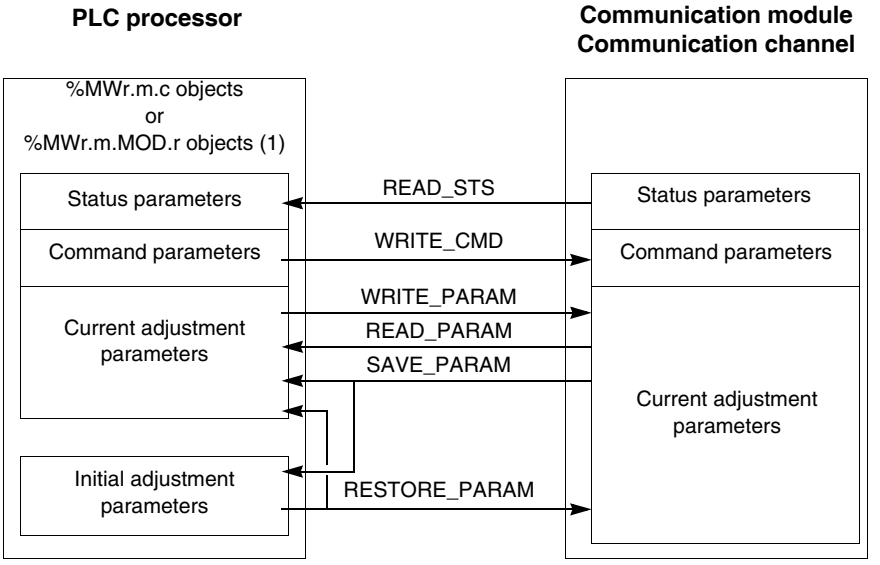
- READ_STS (See Unity Pro, I/O Management Manual, READ_STS) (read status words),
- WRITE_CMD (See Unity Pro, I/O Management Manual, WRITE_CMD) (write command words),
- WRITE_PARAM (See Unity Pro, I/O Management Manual, WRITE_PARAM) (write adjustment parameters),
- READ_PARAM (See Unity Pro, I/O Management Manual, READ_PARAM) (read adjustment parameters),
- SAVE_PARAM (See Unity Pro, I/O Management Manual, SAVE_PARAM) (save adjustment parameters),
- RESTORE_PARAM (See Unity Pro, I/O Management Manual, RESTORE_PARAM) (restore adjustment parameters).

These exchanges apply to a set of %MW objets of the same type (status, commands or parameters) that belong to a channel.

Note: These objects provide information about the module (e.g.: type of channel fault, etc.), can be used to command them (e.g.: switch command) and to define their operating modes (save and restore adjustment parameters in the process of application).

General principle for using explicit instructions

The diagram below shows the different types of explicit exchanges that can be made between the processor and module.



(1) Only with READ_STS and WRITE_CMD instructions.

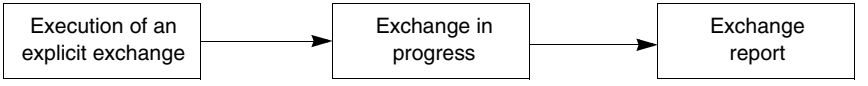
Managing exchanges

During an explicit exchange, it is necessary to check its performance in order that data is only taken into account when the exchange has been correctly executed.

To do this, two types of information is available:

- information concerning the exchange in progress (See *Execution Indicators for an Explicit Exchange: EXCH_STS, p. 168*),
- the exchange report (See *Explicit Exchange Report: EXCH_RPT, p. 168*).

The following diagram describes the management principle for an exchange



Management of Exchanges and Reports with Explicit Objects

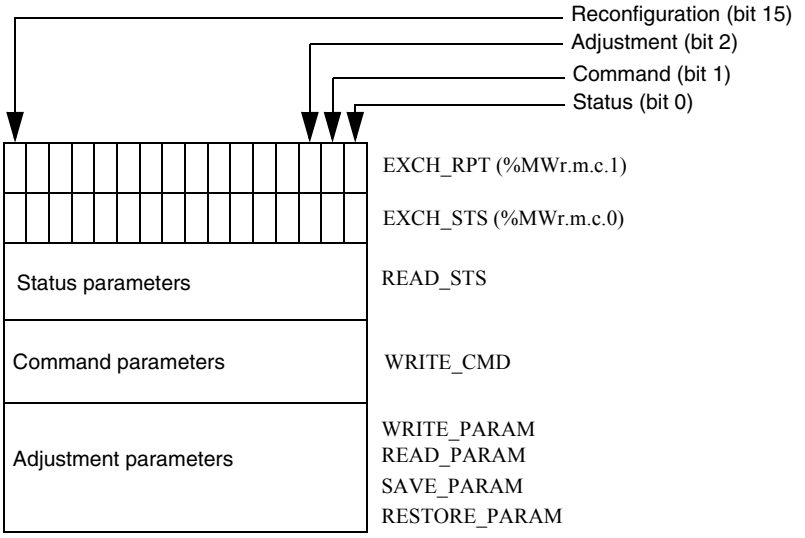
At a Glance

When data is exchanged between the PCL memory and the module, the module may require several task cycles to acknowledge this information. All IODDTs use two words to manage exchanges:

- EXCH_STS (%MWr.m.c.0): exchange in progress,
- EXCH_RPT (%MWr.m.c.1): report.

Illustration

The illustration below shows the different significant bits for managing exchanges:



Description of Significant Bits

Each bit of the words EXCH_STS (%MWr.m.c.0) and EXCH_RPT (%MWr.m.c.1) is associated with a type of parameter:

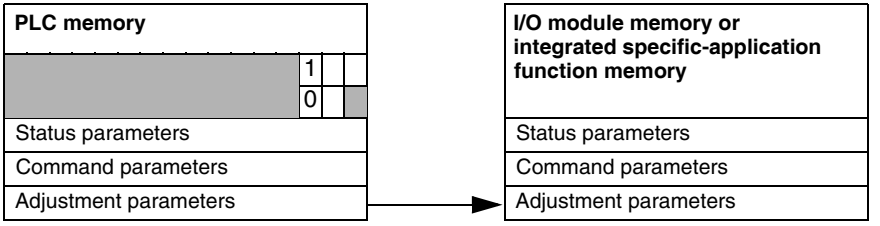
- Rank 0 bits are associated with the status parameters:
  - the STS_IN_PROGR bit (%MWr.m.c.0.0) indicates whether a read request for the status words is in progress,
  - the STS_ERR bit (%MWr.m.c.1.0) specifies whether a read request for the status words is accepted by the module channel.
- Rank 1 bits are associated with the command parameters:
  - the CMD_IN_PROGR bit (%MWr.m.c.0.1) indicates whether command parameters are being sent to the module channel,
  - the CMD_ERR bit (%MWr.m.c.1.1) specifies whether the command parameters are accepted by the module channel.
- Rank 2 bits are associated with the adjustment parameters:
  - the ADJ_IN_PROGR bit (%MWr.m.c.0.2) indicates whether the adjustment parameters are being exchanged with the module channel (via WRITE_PARAM, READ_PARAM, SAVE_PARAM, RESTORE_PARAM),
  - the ADJ_ERR bit (%MWr.m.c.1.2) specifies whether the adjustment parameters are accepted by the module. If the exchange is correctly executed, the bit is set to 0.
- rank 15 bits indicate a reconfiguration on channel c of the module from the console (modification of the configuration parameters + cold start-up of the channel).

Note: r represents the rack number and**m** the position of the module in the rack, while **c** represents the channel number in the module.

Note: Exchange and report words also exist at module level EXCH_STS (%MWr.m.MOD) and EXCH_RPT (%MWr.m.MOD.1) as per IODDT type T_GEN_MOD.

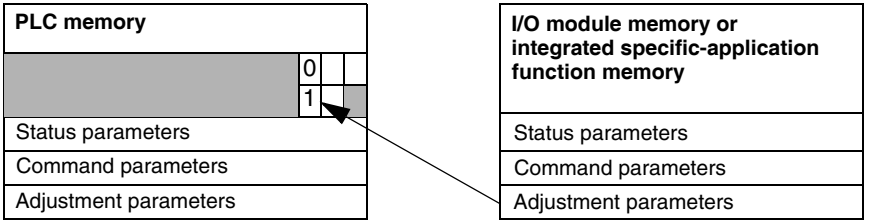
Example

Phase 1: Sending data by using the WRITE_PARAM instruction.



When the instruction is scanned by the PLC processor, the **Exchange in progress** bit is set to 1 in %MWr.m.c.

Phase 2: Analysis of the data by the I/O module and report



When the data is exchanged between the PLC memory and the module, acknowledgement by the module is managed by the ADJ_ERR bit (%MWr.m.c.1.2): Report (0 = correct exchange, 1 = faulty exchange).

Note: There is no adjustment parameter at module level.

Execution Indicators for an Explicit Exchange: EXCH_STS

The table below shows the control bits of the explicit exchanges : EXCH_STS (%MWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading of channel status words in progress	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	R	Command parameters exchange in progress	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Adjust parameters exchange in progress	%MWr.m.c.0.2
RECONF_IN_PROGR	BOOL	R	Reconfiguration of the module in progress	%MWr.m.c.0.15

Note: If the module is not present or is disconnected, explicit exchange objects (Read_Sts for example) are not sent to the module (STS_IN_PROG (%MWr.m.c.0.0) = 0), but the words are refreshed.

Explicit Exchange Report: EXCH_RPT

The table below shows the report bits: EXCH_RPT (%MWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Error reading channel status words (1 = failure)	%MWr.m.c.1.0
CMD_ERR	BOOL	R	Error during a command parameter exchange (1 = failure)	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	Error during an adjust parameter exchange (1 = failure)	%MWr.m.c.1.2
RECONF_ERR	BOOL	R	Error during reconfiguration of the channel (1 = failure)	%MWr.m.c.1.15

11.2 The language objects and generic IODDT applicable to all communication protocols

At a Glance

Aim of this Section

This Section presents the language objects and generic IODDT applicable to all communication protocols.

What's in this Section?

This section contains the following topics:

Topic	Page
Details of IODDT implicit exchange objects of type T_COM_STS_GEN	170
Details of IODDT explicit exchange objects of type T_COM_STS_GEN	171

Details of IODDT implicit exchange objects of type T_COM_STS_GEN

At a Glance

The following table presents the IODDT implicit exchange objects of type `T_COM_STS_GEN` applicable to all communication protocols except Fipio.

Error bit

The table below presents the meaning of the error bit `CH_ERROR` (%Ir.m.c.ERR).

Standard symbol	Type	Access	Meaning	Address
CH_ERROR	EBOOL	R	Communication channel error bit.	%Ir.m.c.ERR

Details of IODDT explicit exchange objects of type T_COM_STS_GEN

At a Glance

This section presents the T_COM_STS_GEN type IODDT explicit exchange objects applicable to all communication protocols except Fipio. It includes the word type objects whose bits have a specific meaning. These objects are presented in detail below.

Sample Variable Declaration: **IODDT_VAR1** of type T_COM_STS_GEN.

Observations

- In general, the meaning of the bits is given for bit status 1. In specific cases an explanation is given for each status of the bit.
- Not all bits are used.

Execution flags of an explicit exchange: EXCH_STS

The table below shows the meaning of channel exchange control bits from channel EXCH_STS (%MWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading of channel status words in progress.	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	R	Current parameter exchange in progress.	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Adjustment parameter exchange in progress.	%MWr.m.c.0.2

Explicit exchange report: EXCH_RPT

The table below presents the meaning of the exchange report bits EXCH_RPT (%MWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Reading error for channel status words.	%MWr.m.c.1.0
CMD_ERR	BOOL	R	Error during command parameter exchange.	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	Error during adjustment parameter exchange.	%MWr.m.c.1.2

Standard channel faults, CH_FLT

The table below shows the meaning of the bits of the status word `CH_FLT` (%MWr.m.c.2). Reading is performed by a **READ_STS**(**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
NO_DEVICE	BOOL	R	No device is working on the channel.	%MWr.m.c.2.0
1_DEVICE_FLT	BOOL	R	A device on the channel is faulty.	%MWr.m.c.2.1
BLK	BOOL	R	Terminal block fault (not connected).	%MWr.m.c.2.2
TO_ERR	BOOL	R	Time out error (defective wiring).	%MWr.m.c.2.3
INTERNAL_FLT	BOOL	R	Internal error or channel self-testing.	%MWr.m.c.2.4
CONF_FLT	BOOL	R	Different hardware and software configurations.	%MWr.m.c.2.5
COM_FLT	BOOL	R	Problem communicating with the PLC.	%MWr.m.c.2.6
APPLI_FLT	BOOL	R	Application error (adjustment or configuration error).	%MWr.m.c.2.7

11.3 Language objects and IODDT's associated to the AS-i bus.

At a glance

Aim of this sub-section

This sub-section presents the language objects and the IODDT's that are associated to the AS-i bus.

What's in this Section?

This section contains the following topics:

Topic	Page
Details of the implicit exchange objects of T_COM_ASI_V1 type IODDTs	174
Details of the explicit exchange objects of the T_COM_ASI_V1 type IODDT	177
Details of implicit exchange objects of the AS-i function	179
Details of explicit exchange language objects for the AS-i function	182
Details of explicit exchange language objects for the AS-i function	184
Details of configuration objects for the AS-i function	186

Details of the implicit exchange objects of T_COM_ASI_V1 type IODDTs

At a Glance

The tables below present the implicit exchange objects of `T_COM_ASI_V1` type IODDTs that apply to AS-i communication.

Error bit

The following table shows the meaning of the `CH_ERROR` (%Ir.m.c.ERR) error bit.

Standard symbol	Type	Access	Meaning	Address
CH_ERROR	EBOOL	R	Communication channel error bit.	%Ir.m.c.ERR

Validity bit

The following table shows the meaning of the `VALID_IN` (%Ir.m.c.0) validity bit.

Standard symbol	Type	Access	Meaning	Address
VALID_IN	EBOOL	R	Indicates that all inputs are valid. Note: when this bit is on 0, it indicates that at least one input is invalid: offline, data exchange off, or channel error mode.	%Ir.m.c.0

List of slaves 0 to 15 faulty

The following table shows the meaning of the bits of word (%IWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_0	BOOL	R	Slave 0 faulty or absent.	%IWr.m.c.0.0
SLAVE_FLT_1	BOOL	R	Slave 1 faulty or absent.	%IWr.m.c.0.1
SLAVE_FLT_2	BOOL	R	Slave 2 faulty or absent.	%IWr.m.c.0.2
SLAVE_FLT_3	BOOL	R	Slave 3 faulty or absent.	%IWr.m.c.0.3
SLAVE_FLT_4	BOOL	R	Slave 4 faulty or absent.	%IWr.m.c.0.4
SLAVE_FLT_5	BOOL	R	Slave 5 faulty or absent.	%IWr.m.c.0.5
SLAVE_FLT_6	BOOL	R	Slave 6 faulty or absent.	%IWr.m.c.0.6
SLAVE_FLT_7	BOOL	R	Slave 7 faulty or absent.	%IWr.m.c.0.7
SLAVE_FLT_8	BOOL	R	Slave 8 faulty or absent.	%IWr.m.c.0.8
SLAVE_FLT_9	BOOL	R	Slave 9 faulty or absent.	%IWr.m.c.0.9
SLAVE_FLT_10	BOOL	R	Slave 10 faulty or absent.	%IWr.m.c.0.10
SLAVE_FLT_11	BOOL	R	Slave 11 faulty or absent.	%IWr.m.c.0.11
SLAVE_FLT_12	BOOL	R	Slave 2 faulty or absent.	%IWr.m.c.0.12
SLAVE_FLT_13	BOOL	R	Slave 13 faulty or absent.	%IWr.m.c.0.13
SLAVE_FLT_14	BOOL	R	Slave 14 faulty or absent.	%IWr.m.c.0.14
SLAVE_FLT_15	BOOL	R	Slave 15 faulty or absent.	%IWr.m.c.0.15

List of slaves 16 to 31 faulty

The following table shows the meaning of the bits of word (%IWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_16	BOOL	R	Slave 16 faulty or absent.	%IWr.m.c.1.0
SLAVE_FLT_17	BOOL	R	Slave 17 faulty or absent.	%IWr.m.c.1.1
SLAVE_FLT_18	BOOL	R	Slave 18 faulty or absent.	%IWr.m.c.1.2
SLAVE_FLT_19	BOOL	R	Slave 19 faulty or absent.	%IWr.m.c.1.3
SLAVE_FLT_20	BOOL	R	Slave 20 faulty or absent.	%IWr.m.c.1.4
SLAVE_FLT_21	BOOL	R	Slave 21 faulty or absent.	%IWr.m.c.1.5
SLAVE_FLT_22	BOOL	R	Slave 22 faulty or absent.	%IWr.m.c.1.6
SLAVE_FLT_23	BOOL	R	Slave 23 faulty or absent.	%IWr.m.c.1.7
SLAVE_FLT_24	BOOL	R	Slave 24 faulty or absent.	%IWr.m.c.1.8
SLAVE_FLT_25	BOOL	R	Slave 25 faulty or absent.	%IWr.m.c.1.9
SLAVE_FLT_26	BOOL	R	Slave 26 faulty or absent.	%IWr.m.c.1.10
SLAVE_FLT_27	BOOL	R	Slave 27 faulty or absent.	%IWr.m.c.1.11
SLAVE_FLT_28	BOOL	R	Slave 28 faulty or absent.	%IWr.m.c.1.12
SLAVE_FLT_29	BOOL	R	Slave 29 faulty or absent.	%IWr.m.c.1.13
SLAVE_FLT_30	BOOL	R	Slave 30 faulty or absent.	%IWr.m.c.1.14
SLAVE_FLT_31	BOOL	R	Slave 31 faulty or absent.	%IWr.m.c.1.15

Details of the explicit exchange objects of the T_COM_ASI_V1 type IODDT

At a Glance

This part presents the explicit exchange objects of `T_COM_ASI_V1` type IODDTs that apply to the AS-i bus. It includes word type objects whose bits have a special meaning. These objects are presented below.

Example of declaration of a variable: **IODDT_VAR1** of type `T_COM_ASI_V1`

Notes

- The meaning of a bit is generally given for the status of the bit when set to 1. In specific cases an explanation is given for each status of the bit.
- Not all bits are used.

Execution indicators for an explicit exchange: EXCH_STS

The table below shows the meaning of channel exchange control bits `EXCH_STS` (%MWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading current channel status words.	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	R	Command parameters exchange in progress.	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Adjust parameters exchange in progress.	%MWr.m.c.0.2

Explicit exchange report: EXCH_RPT

The table below shows the meaning of the exchange report bits `EXCH_RPT` (%MWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Channel status words reading error.	%MWr.m.c.1.0
CMD_ERR	BOOL	R	Command parameter exchange error.	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	Adjust parameter exchange error.	%MWr.m.c.1.2

Note: If the explicit exchange does not last as long as the PLC task cycle time, the %MWr.m.c.0.0 bit never changes to 1.

Standard channel faults, CH_FLT

The table below shows the meanings of the bits of the CH_FLT (%MWr.m.c.2) status word. Reading can be done via a READ_STS (**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT	BOOL	R	One or more device faulty.	%MWr.m.c.2.1
ASI_CONF_FLT	BOOL	R	Physical configuration different from software configuration.	%MWr.m.c.2.3
INTERNAL_FLT	BOOL	R	Internal error or channel self-test.	%MWr.m.c.2.4
CONF_FLT	BOOL	R	Hardware or software configuration fault.	%MWr.m.c.2.5
COM_FLT	BOOL	R	Problem communicating with the PLC.	%MWr.m.c.2.6
APPLI_FLT	BOOL	R	Application fault (adjustment or configuration fault).	%MWr.m.c.2.7

AS-i specific status.

The table below presents the meanings of the (%MWr.m.c.3) status word bits (%MWr.m.c.2). Reading can be done via a READ_STS(**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
ASI_SUPPLY_FLT	BOOL	R	AS-i power supply fault.	%MWr.m.c.3.6

Details of implicit exchange objects of the AS-i function

At a Glance

The tables below present the different implicit exchange word objects. These word objects are not built into `T_COM_ASI_V1` type IODDTs.

Discrete input channel bits

The table below shows the meaning of the discrete input channel bits.

Address	Type	Access	Meaning
%I\b.e\r.m.c	EBOOL	R	Indicates that the input channel (c = 0 to 3) of slave number e (e = 1 to 31) is activated.
Examples:			
%I\b.0\r.m.c	EBOOL	R	Indicates that the input channel (c = 0 to 3) of slave number 0 is activated.
%I\b.31\r.m.c	EBOOL	R	Indicates that the input channel (c = 31 to 3) of slave number 0 is activated.

Discrete output channel bits

The table below shows the meaning of the discrete output channel bits.

Address	Type	Access	Meaning
%Q\b.e\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number e (e = 1 to 31) is activated.
Examples:			
%Q\b.0\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 0 is activated.
%Q\b.31\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 31 is activated.

List of slaves 0 to 15 activated

The following table shows the meaning of the bits of the word (%IWr.m.c.2).

Address	Type	Access	Meaning
%IWr.m.c.2.0	BOOL	R	Slave 0 activated.
%IWr.m.c.2.1	BOOL	R	Slave 1 activated.
%IWr.m.c.2.2	BOOL	R	Slave 2 activated.
%IWr.m.c.2.3	BOOL	R	Slave 3 activated.
%IWr.m.c.2.4	BOOL	R	Slave 4 activated.
%IWr.m.c.2.5	BOOL	R	Slave 5 activated.
%IWr.m.c.2.6	BOOL	R	Slave 6 activated.
%IWr.m.c.2.7	BOOL	R	Slave 7 activated.
%IWr.m.c.2.8	BOOL	R	Slave 8 activated.
%IWr.m.c.2.9	BOOL	R	Slave 9 activated.
%IWr.m.c.2.10	BOOL	R	Slave 10 activated.
%IWr.m.c.2.11	BOOL	R	Slave 11 activated.
%IWr.m.c.2.12	BOOL	R	Slave 12 activated.
%IWr.m.c.2.13	BOOL	R	Slave 13 activated.
%IWr.m.c.2.14	BOOL	R	Slave 14 activated.
%IWr.m.c.2.15	BOOL	R	Slave 15 activated.

List of slaves 16 to 31 activated

The following table shows the meaning of the bits of the word (%IWr.m.c.3).

Address	Type	Access	Meaning
%IWr.m.c.3.0	BOOL	R	Slave 16 activated.
%IWr.m.c.3.1	BOOL	R	Slave 17 activated.
%IWr.m.c.3.2	BOOL	R	Slave 18 activated.
%IWr.m.c.3.3	BOOL	R	Slave 19 activated.
%IWr.m.c.3.4	BOOL	R	Slave 20 activated.
%IWr.m.c.3.5	BOOL	R	Slave 21 activated.
%IWr.m.c.3.6	BOOL	R	Slave 22 activated.
%IWr.m.c.3.7	BOOL	R	Slave 23 activated.
%IWr.m.c.3.8	BOOL	R	Slave 24 activated.
%IWr.m.c.3.9	BOOL	R	Slave 25 activated.
%IWr.m.c.3.10	BOOL	R	Slave 26 activated.
%IWr.m.c.3.11	BOOL	R	Slave 27 activated.
%IWr.m.c.3.12	BOOL	R	Slave 28 activated.
%IWr.m.c.3.13	BOOL	R	Slave 29 activated.
%IWr.m.c.3.14	BOOL	R	Slave 30 activated.
%IWr.m.c.3.15	BOOL	R	Slave 31 activated.

Details of explicit exchange language objects for the AS-i function

At a Glance

The tables below present language objects that provide information on all the slaves present on the AS-i bus. These objects are not built into the IODDT's.

Standard status

The table below shows the meaning of the bit of word %MWr.m.c.2.

Address	Type	Access	Meaning
%MWr.m.c.2.2	BOOL	R	Line error (power supply voltage off or terminal block fault).

AS-i specific status.

The table below provides the meaning of the different bits of the %MWr.m.c.3 word.

Address	Type	Access	Meaning
%MWr.m.c.3.0	BOOL	R	Correct configuration.
%MWr.m.c.3.1	BOOL	R	Slave 0 present.
%MWr.m.c.3.2	BOOL	R	Automatic addressing active.
%MWr.m.c.3.7	BOOL	R	Offline phase active.
%MWr.m.c.3.8	BOOL	R	bit = 1 the AS-i bus is in Data exchange off mode.

List of slaves detected

The table below shows the meaning of the different bits of the %MWr.m.c.4 and %MWr.m.c.5. words.

Address	Type	Access	Meaning
%MWr.m.c.4.n	BOOL	R	n = 0 to 15 -> slaves 0 to 15 respectively have been detected.
%MWr.m.c.5.n	BOOL	R	n = 0 to 15 -> slaves 16 to 31 respectively have been detected.

Configuration of slaves detected

The table below shows the meaning of the different bits of the %MWr.m.c.6 and %MWr.m.c.21 words.

Address	Type	Access	Meaning
%MWr.m.c.6 to %MWr.m.c.21	INT	R	Words 6 to 21 -> slaves 0-1, 2-3, ...,28-29, 30-31, respectively. The least significant bytes refer to the slaves with an even address. The most significant bytes refer to the slaves with an odd address. For each byte bit 0 to 3 = configuration code for input/output (I/O) channels, bit 4 to 7 = identification code (ID).

Parameters of last slave

The table below gives the meaning of the word %MWr.m.c.22.

Address	Type	Access	Meaning
%MWr.m.c.22	INT	R	Contains the answer (value of parameters sent) of the last parametered slave. This is so that the Unity Pro can check the slave has correctly received these values.

Address of last slave

The table below gives the meaning of the word %MWr.m.c.23.

Address	Type	Access	Meaning
%MWr.m.c.23	INT	R	Contains the address of last parametered slave.

Details of explicit exchange language objects for the AS-i function

At a glance

The tables below present the language objects allowing management of the operating mode of the AS-i function and the parameters of the slaves present on the AS-i bus. These objects are not built into the IODDT's.

Operating mode object

This word type object is used to manage the switch to Offline mode (See *AS-i Offline Operating Mode, p. 152*) or Data Exchange Off mode (See *AS-i Data Exchange Off Operating Mode, p. 153*) of the AS-i master.

Note: Use of this object requires a thorough knowledge of AS-i communication principles.

Description

The table below presents the coding of bits 0 to 3 of the %MWr.m.c.24 word providing access to the various modes.

Bit 3	Bit 2	Bit 1	Bit 0	Function
0	0	0	0	Normal Operating Mode
0	0	0	1	Activation of offline mode
0	0	1	0	Deactivation of offline mode
0	0	1	1	Ineffective
0	1	0	0	Activation of data exchange off mode
1	0	0	0	Deactivation of data exchange off mode
1	1	1	1	Ineffective

Note: Offline mode has priority over Data Exchange Off.

Parameter objects

These objects are used to manage the parameters of slave devices. They can be modified without stopping the AS-i function.

Description

The table below provides the meaning of the different bits of words %MWr.m.c.25 to %MWr.m.c.56.

Number	Type	Access	Indicates
%MWr.m.c.25	INT	R/W	Not significant.
%MWr.m.c.26 to 56	INT	R/W	Contain the parameter values of slaves 1 to 31 respectively.

Details of configuration objects for the AS-i function

At a Glance

%KWr.m.c.d. configuration constants are accessible in read-only mode and correspond to the configuration parameters entered using the Configuration editor.

Configuration objects

The table below shows constant objects on the AS-i bus.

Address	Type	Access	Meaning
%KWr.m.c.0	INT	R	When byte 0 = 10, this shows that AS-i is master.
%KWr.m.c.1.n	BOOL	R	n = 0 to 15 -> slaves 0 to 15 respectively.
%KWr.m.c.2.n	BOOL	R	n = 0 to 15 -> slaves 16 to 31 respectively.
%KWr.m.c.3 to %KWr.m.c.18	INT	R	Words 3 to 18 -> slaves 0 to 31 respectively. Byte 0: bit 0 to 3 = configuration code for input/output (I/O) channels of slave 0, bit 4 to 7 = identification code (ID) of slave 0, Byte 1: bit 0 to 3 = configuration code for input/output (I/O) channels of slave 1, bit 4 to 7 = identification code (ID) of slave 1.
%KWr.m.c.19	INT	R	Byte 0: bit 0 = 0 if the output fallback is set to 0, bit 0 = 1 if outputs are maintained on fallback, bit 1 = 0 automatic addressing disabled, bit 1 = 1 automatic addressing enabled.
%KWr.m.c.20 to %KWr.m.c.51	INT	R	Catalog information: word 20: catalogue identifier slave 0, word 51: catalogue identifier slave 31.

11.4 The IODDT type T_GEN_MOD applicable to all modules

Details of the Language Objects of the IODDT of Type T_GEN_MOD

At a Glance

All the modules of Premium PLCs have an associated IODDT of type T_GEN_MOD.

Observations

- In general, the meaning of the bits is given for bit status 1. In specific cases an explanation is given for each status of the bit.
- Not all bits are used.

List of Objects

The table below presents the objects of the IODDT:

Standard symbol	Type	Access	Meaning	Address
MOD_ERROR	BOOL	R	Module error bit	%Ir.m.MOD.ERR
EXCH_STS	INT	R	Module exchange control word.	%MWr.m.MOD.0
STS_IN_PROGR	BOOL	R	Reading of status words of the module in progress.	%MWr.m.MOD.0.0
EXCH_RPT	INT	R	Exchange report word.	%MWr.m.MOD.1
STS_ERR	BOOL	R	Fault when reading module status words.	%MWr.m.MOD.1.0
MOD_FLT	INT	R	Internal error word of the module.	%MWr.m.MOD.2
MOD_FAIL	BOOL	R	Internal error, module failure.	%MWr.m.MOD.2.0
CH_FLT	BOOL	R	Faulty channel(s).	%MWr.m.MOD.2.1
BLK	BOOL	R	Terminal block fault.	%MWr.m.MOD.2.2
CONF_FLT	BOOL	R	Hardware or software configuration fault.	%MWr.m.MOD.2.5
NO_MOD	BOOL	R	Module missing or inoperative.	%MWr.m.MOD.2.6
EXT_MOD_FLT	BOOL	R	Internal error word of the module (Fipio extension only).	%MWr.m.MOD.2.7
MOD_FAIL_EXT	BOOL	R	Internal fault, module unserviceable (Fipio extension only).	%MWr.m.MOD.2.8
CH_FLT_EXT	BOOL	R	Faulty channel(s) (Fipio extension only).	%MWr.m.MOD.2.9
BLK_EXT	BOOL	R	Terminal block fault (Fipio extension only).	%MWr.m.MOD.2.10
CONF_FLT_EXT	BOOL	R	Hardware or software configuration fault (Fipio extension only).	%MWr.m.MOD.2.13
NO_MOD_EXT	BOOL	R	Module missing or inoperative (Fipio extension only).	%MWr.m.MOD.2.14

Implementation of AS-i V2 bus software

IV

At a Glance

Subject of this part

This part presents the software implementation of the AS-i V2 bus with Unity Pro software.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
12	Principles of the software implementation of the AS-i V2 Bus	191
13	AS-i V2 bus configuration	203
14	Debugging the AS-i V2 bus	223
15	AS-i V2 operating mode	239
16	AS-i V2 performance	245
17	DFB for AS-i Security Monitor	247
18	Language objects of the AS-i V2 bus	257

Principles of the software implementation of the AS-i V2 Bus

12

At a glance

Aim of this Chapter

This chapter presents the principles of software implementation of the AS-i V2 bus.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Presentation of the installation of the AS-i V2 bus	192
Architecture of the TSX SAY 1000 Module	194
Structure of a Standard Address AS-i	196
Structure of a Standard Address AS-i slave	198
Addressing of language objects associated with the slave devices connected to the AS-i bus.	200

Presentation of the installation of the AS-i V2 bus

Introduction

The AS-i V2 bus (Actuator Sensor-Interface Version 2) allows the interconnection on a single cable of sensor devices/actuators at the lowest level of automation. These sensors/actuators will be defined in the documentation as slave devices.
To implement the AS-i application you need to define the physical context of the project into which it will be built (rack, power supply, processor, modules, AS-i slave devices connected to the bus) then ensure its software implementation.
The software installation of the application modules is performed from the various Unity Pro editors:

- in local mode,
- in online mode.

The following is the recommended order of the installation phases, but it is possible to modify the order of some phases (for example start with the configuration phase).

AS-i V2 Bus

The AS-i V2 functions are built into the **TSX SAY 1000** module. The AS-i bus then allows :

- up to 62 standard and/or extended address slaves,
- up to 248 inputs and 186 outputs,
- up to 124 analog values integrated in the master AS-i,
- a cycle time of 10ms maximum.

AS-i V2 bus installation principle

The following table shows the different implementation phases of the AS-i V2 :

Phase	Description	Mode
Variable Declarations	Declaration of variable of the IODDT type for the project modules and variables.	Local (1)
Programming	Programming the project and functions carried out using the AS-i V2 bus.	Local (1)
Configuring	Declaration of modules and slave devices.	Local
	Configuring the module channels.	
	Entering configuration parameters.	
Association	Association of IODDT's with the configured modules (variables editor).	Local (1)
Generation	Generation (analysis and editing of links) of the project.	Local
Transfer	Transfer of the project to the PLC.	Connected
Adjust and Debug	Debugging the project from debug screens/animation tables.	Connected
	Modification of the program and the adjustment parameters.	
Documentation	Building the documentation file and printing of the various information concerning the project.	Connected (1)
Operation/ Diagnostics	Display of the various information necessary for conducting the project.	Connected
	Project and modules diagnostics.	
Legend		
(1)	These various phase can also be performed in the other mode.	

Architecture of the TSX SAY 1000 Module

At a glance

The **TSX SAY 1000** module operates according to the master/slave mode. The master only controls exchanges on the bus.
The AS-i standard sets several operating levels offered by the master:

- Profile M0 and M0e - Minimum Master: the master only puts forward the configuration of slaves connected to the bus on power-up, and only input/output exchanges.
- Profile M1 and M1e - Full Master: this profile covers all the operating functions set by the AS-i standard,
- Profile M2 and M2e - Reduced Master: this profile corresponds to profile M0 operating functions with a slave-parametering option.

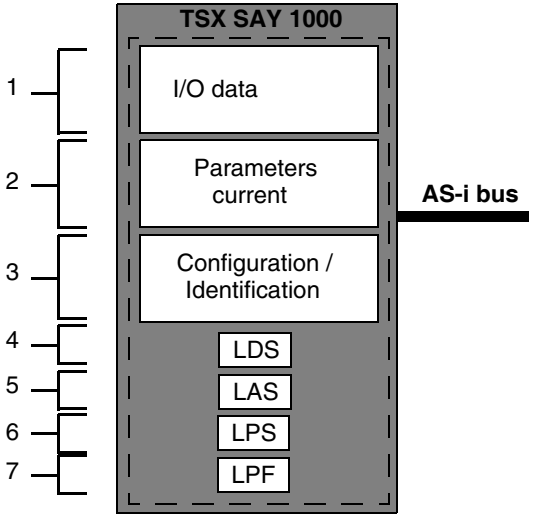
The master profiles with an "e" support extended profiles.

Note: The **TSX SAY 1000** module has an M2e profile, with the additional option of reading slave and slave channel diagnostics information.

The module includes data fields which are used to manage the lists of slaves and I/O data images. This information is stored in a volatile memory.

Illustration of the architecture

The figure below shows the **TSX SAY 1000** module architecture.



Description of constituent elements

The table below shows the different elements that make up the architecture of the **TSX SAY 1000** module.

Number	Item	Description
1	I/O data	Images of 248 inputs and 186 outputs of AS-i V2 bus.
2	Current parameters	Image of parameters of all slaves.
3	Configuration/ Identification	This field contains all the I/O codes and identification codes for all the detected slaves.
4	LDS	List of all the slaves detected on the bus.
5	LAS	List of active slaves on the bus.
6	LPS	List of slaves provided on the bus and configured by Unity Pro.
7	LPF	List of slaves having a device fault.

Structure of a Standard Address AS-i

At a glance

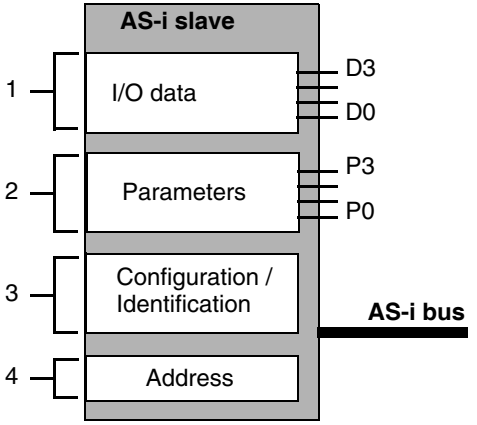
The AS-i V2 bus is used to interconnect 31 standard address devices.
The standard address slaves each have:

- 4 input bits,
- 4 output bits,
- 4 parametering bits.

The AS-i V2 bus can manage a maximum of 124 standard address slave Inputs and 124 Outputs.
Each slave has its own address and a profile (defines variables exchange).

Structure illustration

The figure below shows the structure of a standard address slave.



Description of constituent elements

The table below shows the different elements that make up the structure of a standard address slave.

Number	Item	Description
1	Input/output data	Input data is stored by the slave and made available for the AS-i master. Output data is updated by the master module.
2	Parameters	The parameters are used to control and switch internal operating modes to the sensor or the actuator.
3	Configuration/ Identification	This field contains: ● the code which corresponds to I/O configuration, ● the slave identification codes (ID, ID1, ID2).
4	Address	Physical address of slave.
Note: The operating parameters, address, configuration and identification data are saved in a non-volatile memory.		

Structure of a Standard Address AS-i slave

At a glance

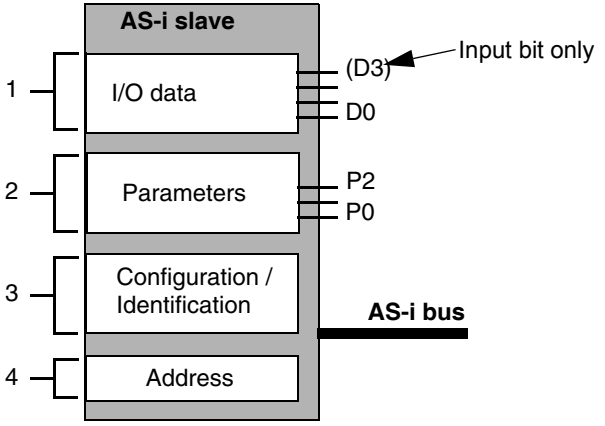
The extended AS-i V2 bus is used to interconnect 62 extended address devices.
The extended address slaves each have:

- 4 input bits,
- 3 output bits,
- 3 parametering bits.

The AS-i V2 bus can manage a maximum of 248 Inputs and 186 Outputs from extended address slaves.
Each slave has its own address, profile and sub-profile (defines variables exchange).

Structure illustration

The figure below shows the structure of an extended address slave.



Description of constituent elements

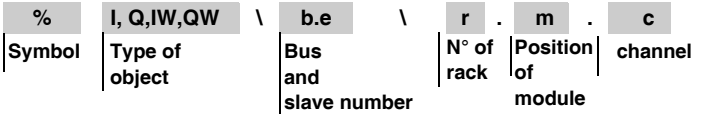
The table below shows the different elements that make up the structure of an extended address slave.

Number	Item	Description
1	Input/output data	Input data is stored by the slave and made available for the AS-i master. Output data is updated by the master module.
2	Parameters	The parameters are used to control and switch internal operating modes to the sensor or the actuator.
3	Configuration/ Identification	This field contains: ● the code which corresponds to I/O configuration, ● the slave identification (ID) code, ● the slave identification codes (ID1 and ID2).
4	Address	Physical address of slave.
Note: The operating parameters, address, configuration and identification data are saved in a non-volatile memory.		

Addressing of language objects associated with the slave devices connected to the AS-i bus.

At a glance

Acquisition of inputs and updates of slave device outputs connected to the AS-i bus are carried out automatically. This occurs at the start and end of each cycle respectively of the task in which the **TSX SAY 100** module is configured.
The user program has access to these inputs and outputs via the language objects. Addressing is defined in the following way:



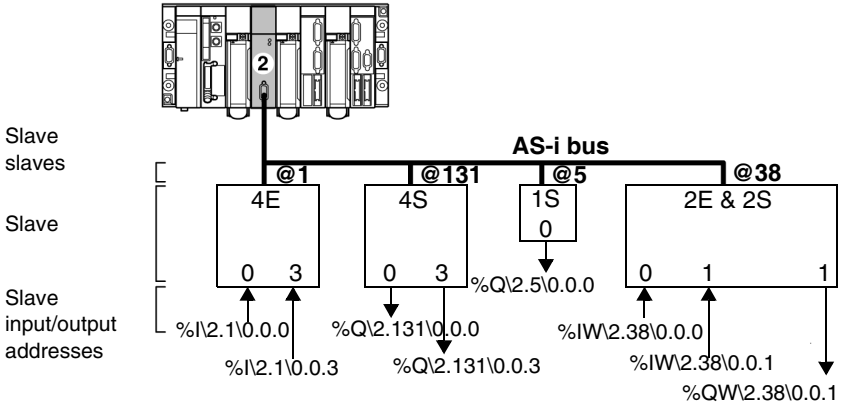
Syntax

The table below describes the different elements that make up addressing.

Family	Item	Values	Indicates
Symbol	%	-	-
Object type	I Q IW QW	- -	Picture of the All or Nothing input of the module, Picture of the All or Nothing output of the module, Picture of the analog input of the module, Picture of the analog output of the module, This information is exchanged automatically each cycle of the task to which they are attached.
Bus and slave No.	b	2 to 999	Bus number.
	e	1 to 31 101 to 131	Slave number for bank A. Slave number for bank B (offset of 100 in relation to the slave number of bank A).
Rack number	r	0	Virtual rack number.
Module position	m	0	Virtual module position
Channel	c	0 to 6	Input or output number

Example

%I\2.1\0.0.3 indicates: input 3 of bus number 2 of slave TOR /A 1 and implicitly rack 0 and module 0 of the **TSX SAY 1000** module.
%Q\2.131\0.0.0 indicates: output 0 of bus number 2 of slave TOR /B 131 and implicitly rack 0 and module 0 of the **TSX SAY 1000** module.
%IW\2.38\0.0.0 indicates: input 0 of bus number 2 of slave ANA / A 38 and implicitly rack 0 and module 0 of the **TSX SAY 1000** module.
Illustration:



Note: The physical address of an AS-i slave is programmed by a console.
An analog slave is configured only on slot /A.
The number of an extended TOR /B slave lies between 101 and 131.
The number of a standard TOR /A slave, or an ANA slave (which is always standard) lies between 1 and 31.
When a standard address setting slave is set at /A, an extended address setting slave at /B cannot have the same address. Only two extended address setting slaves can have the same address at /A and /B.

Multiple Address Settings

When one or more slaves are connected, make sure that you do not assign an address which is already being used by a slave on the bus.
If slave address is doubled then two scenarios may occur :

- the two slaves with identical addresses have the same profile and manage identical I/O's: the AS-i master bus does not detect any error.
- the two slaves with identical addresses manage different I/O's: the AS-i master bus can detect transmission errors when accessing the I/O from one of the two slaves.

AS-i V2 bus configuration

13

At a Glance

Subject of this Chapter

This Chapter describes the Configuration aspect for installing the AS-i V2 bus.

What's in this Chapter?

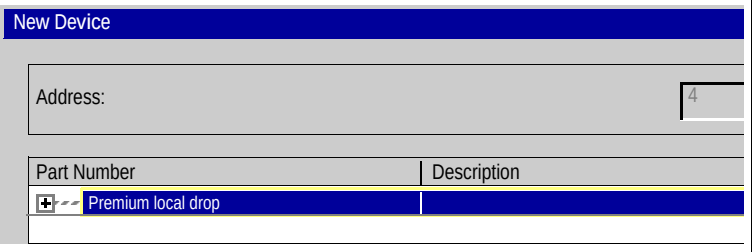
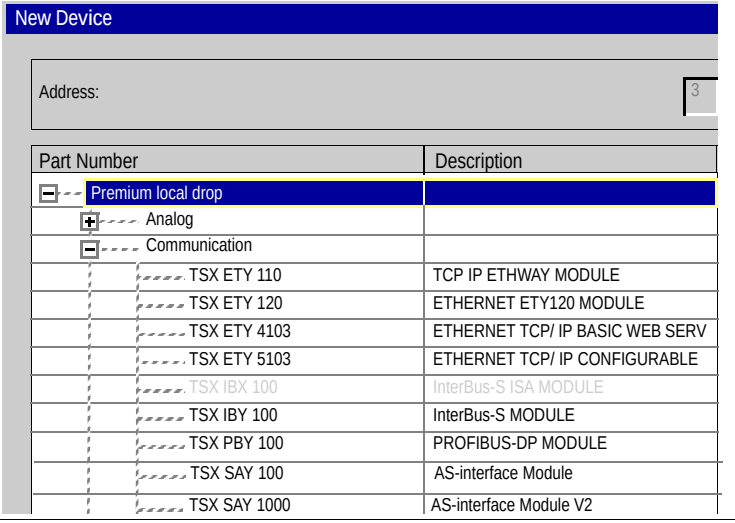
This chapter contains the following topics:

Topic	Page
How to declare an AS-i V2 communication module in the PLC Rack	204
Description of the configuration screen of an AS-i V2 communication module	205
How to define a slave device on the AS-i V2 bus	207
AS-i V2 Bus Displayed in the Project Browser	211
How to modify the AS-i V2 Bus software configuration	213
How to access the description of an AS-i V2 slave	214
How to define a new slave profile in the standard AS-i V2 catalog	216
How to modify the general parameters of an AS-i V2 slave: Automatic addressing	218
How to modify the parameters of an AS-i V2 slave	219
How to modify the parameters of an analog AS-i V2 slave	220
How to Modify the Parameters of an AS-i V2 Slave with Combined Parameters	221
AS-i V2 security device issues	222

How to declare an AS-i V2 communication module in the PLC Rack

Procedure

This operation makes it possible to insert an AS-i communication module in the PLC rack.

Step	Action
1	Open the hardware configuration editor.
2	Select the slot where you wish to insert the module.
3	Select the command **New Device** from the contextual menu. **Result**: the **New Device** window appears. 
4	Open up the line **Premium local input/output station** and the line **Communication** by clicking on the + sign. **Result** : 
5	Select the module **TSX SAY 1000** then confirm with the command **OK**.

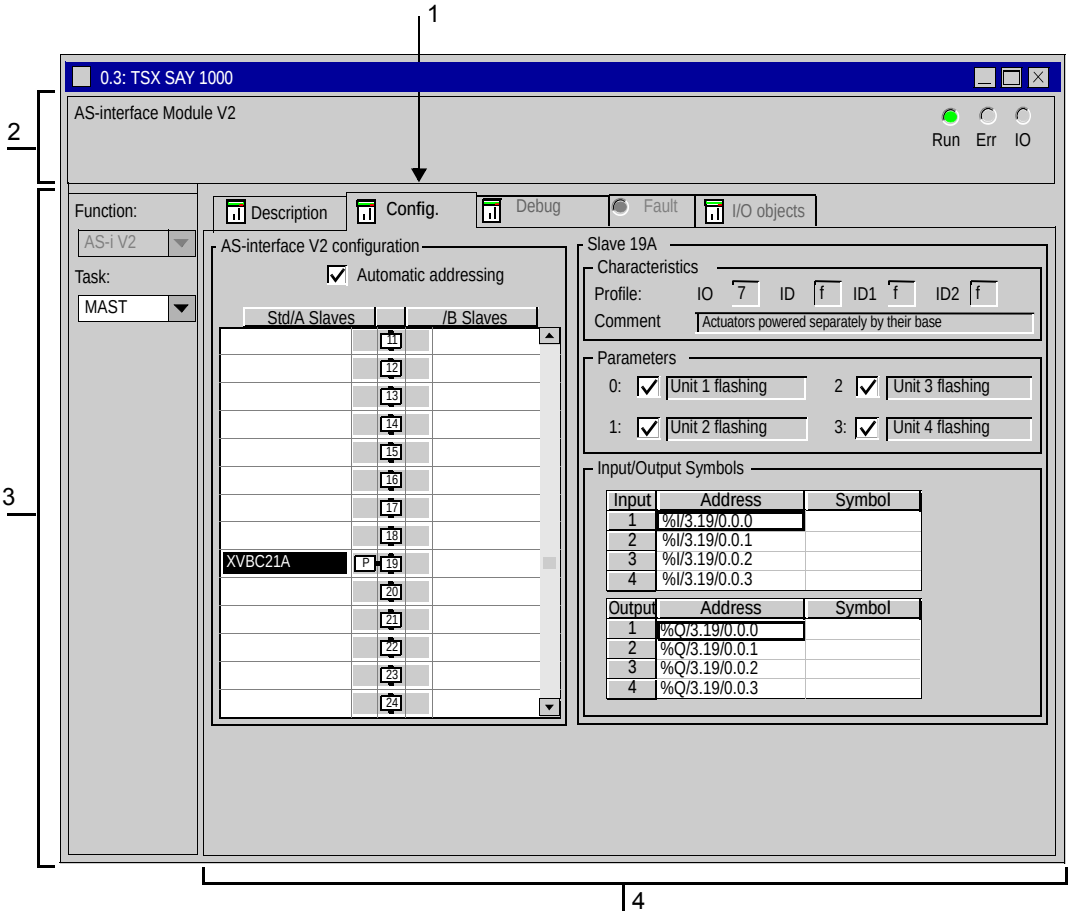
Description of the configuration screen of an AS-i V2 communication module

At a Glance

The configuration screen of the AS-i V2 module gives access to the parameters associated with the module and the slave devices.

Illustration

The figure below represents a configuration screen.



Description

The following table shows the different elements of the configuration screen and their functions.

Address	Item	Function
1	Tabs	The front tab indicates the mode in progress (**Debug** for this example). Each mode can be selected with the corresponding tab. Possible modes are: • **Configuration**, • **Debug** (or diagnostics), only accessible in online mode, • **Diagnostics** (Default), accessible only in Online mode. The tab **I/O Objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) is used to presymbolize the input/output objects.
2	**module**zone	Specify the abbreviated title of the module and the status of the module in online mode There are three indicators in the same zone that provide the status of module in online: • **RUN** indicates the operating mode of the module. • **ERR** signals an internal fault in the module. • **I/O** signals an external fault in the module or an application fault.
3	**General parameters**zone	makes it possible to select the general parameters associated with the channel : • **Function :** the AS-i 2 bus function is frozen (grayed). • **Task:** defines the **MAST** or **FAST** task through which the channel's implicit exchange objects will be exchanged.
4	**configuration**field	Makes it possible to configure the channel configuration parameters. Certain choices may be frozen and appear grayed. It is broken down into four fields: • AS-interface configuration, • the selected slave characteristics, • the parameters applied to the chosen slave, • the input and output symbols associated with a device.

How to define a slave device on the AS-i V2 bus

At a Glance

Unity Pro software offers a catalog of Schneider products which groups together all of the AS-i V2 slaves that are available. This catalog is currently structured according to the following families :

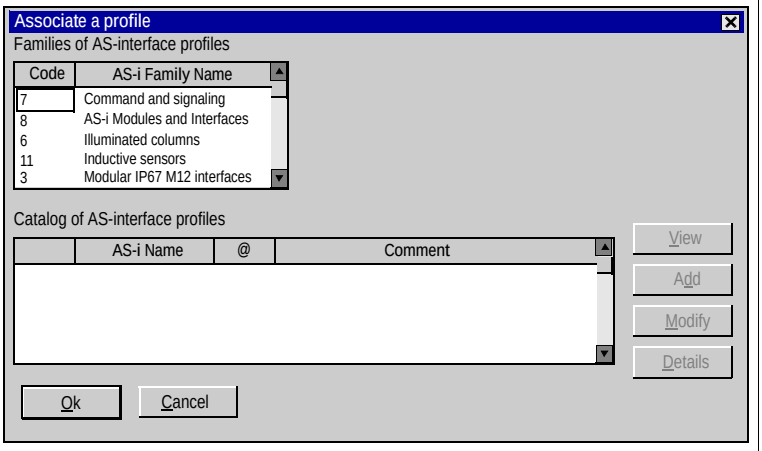
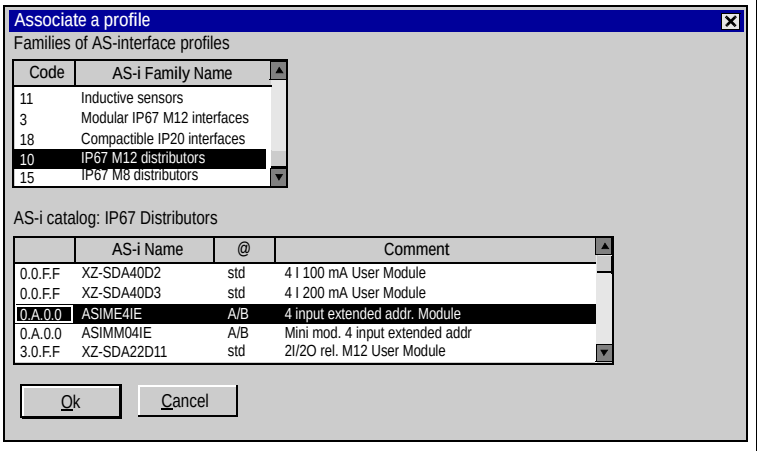
- Keyboards
- Illuminated columns
- Command and signaling
- AS-i modules and interfaces
- Motor-starters
- Inductive sensors
- Phototronic sensors
- Private family
- IP20 linkage interfaces
- Position switches
- Analog modules
- Security modules
- Logical modules
- IP67 Distributors
- Variable speed controllers

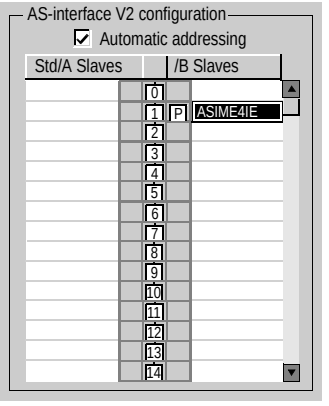
Selecting **Private family** enables the user to enhance the Unity Pro catalog with specific AS-i products via his programming terminal.

Note: A project using AS-i products from the **Private Family** catalog is always linked to the usage of the same **Private Family** catalog of the workstation on which the catalog was created.

Procedure

This operation enables the declaration of a slave device on the AS-i V2 bus.

Step	Action
1	Access the AS-i module's hardware configuration screen.
2	In the **AS-interface V2 configuration** field, double click in the cell corresponding to the new slave's host slot number (1A to 31A or 1B to 31B) or select this cell then execute the command **Edit** → **Add an AS-i slave**. **Result**: The screen **Associate a profile** appears. 
3	In the **AS-i Family Name** field select the required family. **Result**: The **Profile catalog** linked to the selected family appears.  Column Legend @ : std: standard address slave (for configuration in column /A). A/B: extended address slave (for configuration in columns /A or /B).

Step	Action
4	In the **Profile catalogue** select the required device.
5	Confirm the selection with **OK**. **Result**: The slave device is defined in its slot, the reference of the connected device appears opposite the number of the slave. 
6	To connect other slave devices to the AS-i V2 bus, repeat the procedure from step 2.

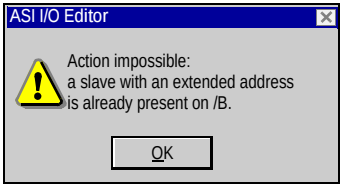
Configuration Rules

The configuration rules for the slaves are as follows :

- The **/B Slaves** column can only support extended address slaves, and this on condition that the **std /A Slaves** cell immediately to its left is not occupied by a standard address slave.
- The **std /A Slaves** column can support extended address slaves. This column can also support standard address slaves, as long as the **/B Slaves** cell immediately to its right is not occupied by an extended address slave.

A maximum of 62 extended or 31 standard address slaves can be configured.

A validity check is carried out during slave entry. A message like the one below will be displayed if this is not possible:



Note: An analog slave must be configured as a standard address slave (in the **/A Slaves** column) and prohibits the use of the slot immediately to its right (in the **/B Slaves** column).

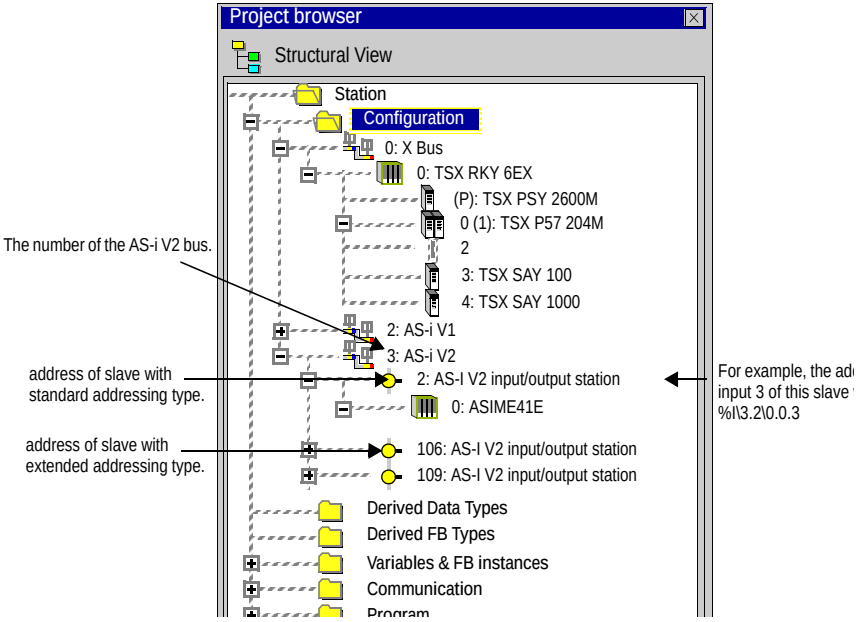
AS-i V2 Bus Displayed in the Project Browser

At a Glance

When you declare a **TSX SAY 1000** module on the PLC rack, the AS-i v2 bus is represented in the **configuration** directory of the project browser. The number of the AS-i V2 bus is calculated automatically by Unity Pro. **This value may be modified**.

After having **declared** all the slaves on the AS-i V2 bus and **validated** the configuration, the AS-i V2 slaves also appear on the AS-i V2 bus of the project browser. Each slave appears with its address number. Displaying the AS-i V2 bus and slaves enables you to know at a glance their topological addressing.

The following illustration shows the AS-i V2 bus and slaves in the project browser;



In the project browser, the AS-i V2 bus is represented by **AS-i V2** and the AS-i bus is represented by **AS-i V1**.

Modifying the Bus Number

It is possible to modify the AS-I bus number by following the below instructions:

Step	Action
1	Close the AS-I module configuration screen (if it is open).
2	In the project browser, right click on the AS-I module and select properties.
3	In the **Bus properties** window, select the desired bus number (between 1 and 999).
4	Click on **OK** to confirm the modification. **Result:** the addresses of the slave devices directly associated with the AS-I bus module are automatically modified.

Note: You are advised not to use the number 2 bus for the AS-I bus because this number is reserved for the Fipio bus.

How to modify the AS-i V2 Bus software configuration

At a Glance

From the AS-i V2 module's configuration screen, Unity Pro software offers a group of functions which allow you to easily modify, in local mode, the software configuration of the AS-i V2 bus.

Note: standard Windows keyboard shortcut keys (Del, Ctrl-X, Ctrl-C, Ctrl-V) are also available for the following operations:

Procedure for deleting a slave

This operation allows the deletion of a slave declared on an AS-i Bus.

Step	Action
1	Select the slave to be deleted.
2	Select the command**Edit** → **Delete an AS-i slave**.

Procedure for moving a slave

This operation allows a slave declared on an AS-i V2 Bus to be moved.

Step	Action
1	Select the slave to be moved.
2	Select the command**Edit** → **Cut an AS-i slave**.
3	Select the new required slot.
4	Select the command **Edit** → **Paste an AS-i slave**.

Procedure for copying a slave

This operation allows the copying of a slave declared on an AS-i V2 Bus.

Step	Action
1	Select the slave to be copied.
2	Select the command **Edit** → **Copy an AS-i slave**.
3	Select the slot of the new slave.
4	Select the command **Edit** → **Paste an AS-i slave**.

How to access the description of an AS-i V2 slave

At a Glance

The Unity Pro software allows access to all the information relating to an AS-i V2 device such as:

- the definition of a profile,
- the details of a profile.

Definition of a profile

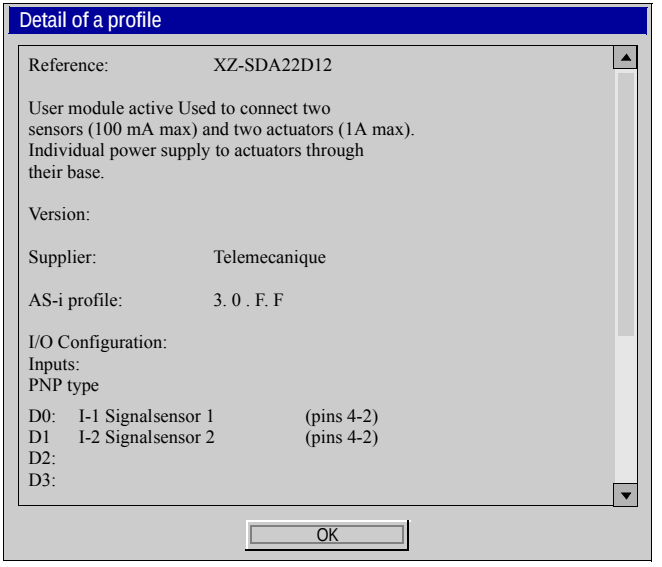
A profile is defined by:

- its name,
- a comment (optional),
- identifiers (IO, ID, ID1, ID2),
- a number of inputs and/or outputs,
- operation parameters.

Note: profile description for **Private Family** products is not accessible.

Details of a profile

The **Details** function allows access, for a given slave, to all the information presented in the catalog file.


Detail of a profile
Reference: XZ-SDA22D12
User module active Used to connect two sensors (100 mA max) and two actuators (1A max). Individual power supply to actuators through their base.
Version:
Supplier: Telemecanique
AS-i profile: 3. 0 . F. F
I/O Configuration:
Inputs:
PNP type
D0: I-1 Signalsensor 1 (pins 4-2)
D1 I-2 Signalsensor 2 (pins 4-2)
D2:
D3:
OK

Procedure for accessing information on a profile

The following table shows the procedure for displaying the characteristics of a slave device.

Step	Action
1	Access to the hardware configuration screen of the AS-i V2 module.
2	Double click on the required slave. **Result:** The window**Associate a profile** displays and highlights the device in question
3	Select the profile family and the reference for the desired slave.
4	Click on the button: ● **View** to access definition information, ● **Details** to access all the information.

How to define a new slave profile in the standard AS-i V2 catalog

At a Glance

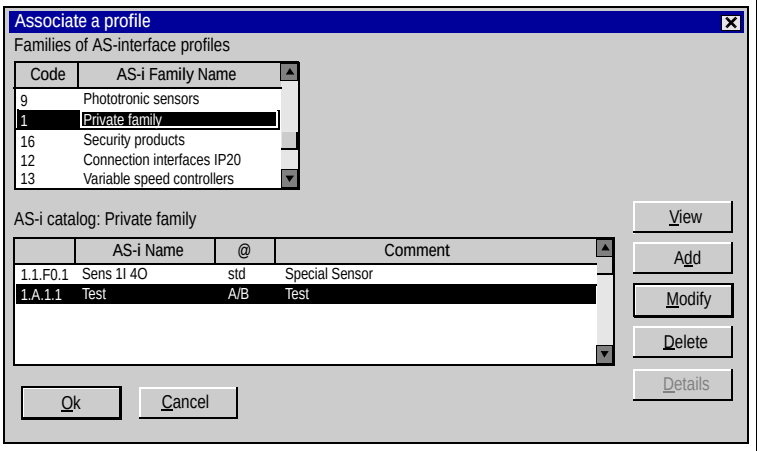
The Unity Pro software offers the possibility of defining a slave profile, which was not provided, in the standard catalog.

The new profile, defined in this way, is added to the catalogue in **Private family**.

This profile can then be used as a standard catalogue profile.

Procedure

The following table presents the procedure for defining a slave profile, which is not provided in the standard catalogue.

Step	Action
1	Access the AS-i module's hardware configuration screen.
2	Double click in a slave's host cell (1 to 31 /A or /B). **Result**: The screen**Associate a profile** appears.
3	Select **Private family** in the field **AS-i Family Name**. **Result**: The **AS-i catalog** linked to the selected family appears. 
4	Click on the button**Add**. A new **AS-interface profile definition** window opens.
5	Enter: ● the name of the new profile, ● a comment (optional).
6	Select: ● the **IO** code (corresponds to the input/output configuration), ● the **ID** code (identifier) (plus ID1 and for an extended type),
7	For each parameter define: ● the system's acknowledgement (box checked), ● a label (optional).
8	Confirm the introduction of a new profile using **Confirm**.

How to modify the general parameters of an AS-i V2 slave: Automatic addressing

At a Glance

Each slave on the AS-i bus must be assigned (via configuration) a unique physical address. This must be the same as the one declared in Unity Pro.

The Unity Pro software offers an automatic slave addressing utility so that an AS-i console does not have to be used.

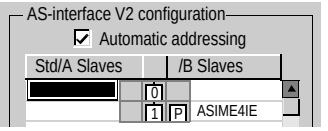
The automatic addressing utility, which can be accessed via Unity Pro, is used for:

- replacing a faulty slave (See *Automatic replacement of a faulty AS-i V2 slave, p. 236*),
- inserting a new slave (See *How to insert a slave device into an existing AS-i V2 configuration., p. 237*).

Note: A new configuration with automatic addressing will not be accepted if one or more slaves with a 0 address are on the bus. In this case, the `Configuration refused by module` message appears.

Procedure

The table below shows the procedure for setting the **Automatic addressing** parameter.

Step	Action
1	Access the AS-i V2 communication module's configuration screen.
2	Click on the **Automatic addressing** check box found in the **AS-interface V2 configuration** zone. **Result**: The **Automatic addressing** utility will be activated (box checked) or disabled (box not checked.  **Note**: By default, the **Automatic addressing** parameter has been selected in the configuration screen.

How to modify the parameters of an AS-i V2 slave

At a Glance

The **Parameters** zone of the configuration screen enables the user to choose whether or not to activate the parameters of certain slaves.

The parameters displayed differ according to the slave in use, please refer to slave documentation for further details.

Example of parameters

On certain slaves, the following may be activated:

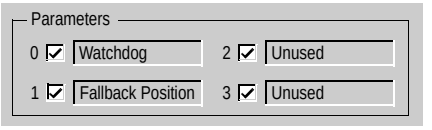
- The internal **watchdog**, upon a communication stop with the (TSX SAY 1000) bus master.
- The output **fallback position** pre-programmed in the slave.
  This parameter activates the slaves' output fallback when changing to STOP or when the PLC is faulty.

The fallback position is pre-defined in the slave device. Possible modes are:

- **Fallback to 0**: the outputs of AS-i slaves present on the bus are set to 0 status.
- **Maintain status**: AS-i slave outputs remain in the state they were before STOP mode was engaged.

Procedure

The table below shows the procedure for selecting the **Watchdog** and the **Fallback Position** assigned to the output of a slave supporting these parameters.

Step	Action
1	Access the AS-i V2 communication module's configuration screen.
2	Select the slave whose parameters are to be modified.
3	Click on the **Watchdog** checkbox and/or **Fallback Position** checkbox located in the **Parameters** zone. **Result**: The watchdog and/or fallback position will be activated in the slave device. 

How to modify the parameters of an analog AS-i V2 slave

At a Glance

The **Parameters** zone of the configuration screen is used to choose whether to activate or deactivate pre-defined parameters in an analog slave. For example:

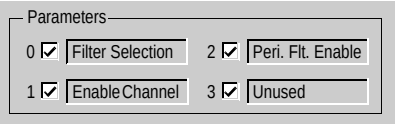
- **Filter Selection**, which activates the rejection of 50/60 Hz on inputs.
- **Channel 2 Enabling**, which enables the second analog channel.
- **Peri Flt Enabling**, which enables peripheral faults to be displayed.

The parameters displayed differ according to the slave in use. Please refer to slave documentation for further details.

Note: The module **TSX SAY 1000** can only manage analog slaves of the type **7.3**.

Procedure

The following table shows the procedure for defining the parameters of an analog slave device.

Step	Action
1	Access the AS-i V2 communication module's configuration screen.
2	Select the slave whose parameters are to be modified.
3	Click on the **Filter Selection** and/or **Channel 2 Enabling** and/or **Peri Flt Enabling** checkboxes located in the parameters **Parameters** zone. **Result**: The parameters checked will be activated in the slave device. 

How to Modify the Parameters of an AS-i V2 Slave with Combined Parameters

At a Glance

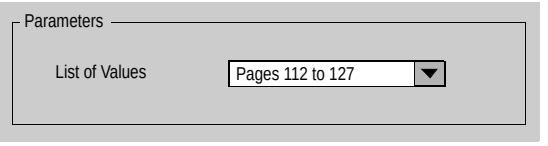
Some AS-i V2 slaves have several pre-defined combinations of parameters. The user can then choose the combination of parameters required from a list.

The **Parameters** zone of the configuration screen allows the user to choose this combination of parameters (called Page).

As the parameters differ according to the slave in use, please refer to slave documentation for further details.

Procedure

The table below shows the procedure for selecting the parameters of the slave devices with combined parameters.

Step	Action
1	Access the AS-i V2 communication module's configuration screen.
2	Select the slave whose parameters are to be modified.
3	Select a combination of parameters (page) in the **List of values** zone. **Result**: The list of parameters selected will be activated in the slave device. 

AS-i V2 security device issues

At a Glance

The **TSX SAY 1000** module supports AS-i security devices on its bus. The addresses of these devices can be configured by the Unity Pro software.

The AS-i security product consists of a Monitor and one or more slaves. The bus master sees these devices as standard address slaves, but with a special profile. There is a **Security Module** family on the Configuration Screen (See *How to define a slave device on the AS-i V2 bus, p. 207*).

Precautions of Use

Input/Output objects from these devices **should not** be used in the project program, as their values are not significant.

The diagnostics information on the Security Monitor may, however, be utilized by the DFB supplied with this product.

Debugging the AS-i V2 bus

14

At a Glance

Aim of this chapter

This Chapter describes the Debug aspect of the AS-i V2 bus.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Introduction to the Debug function	224
Description of an AS-i V2 module's debug screen	225
How to access module- and channel-diagnostics functions on an AS-i V2 device.	227
Displaying the slaves' status	229
How to access the adjustment of the parameters of an AS-i V2 device	231
How to access the AS-i V2 digital channels' forcing/unforcing function	233
How to access the SET and RESET commands of the AS-i V2 digital channels	234
How to modify the value of an analog channel	235
Automatic replacement of a faulty AS-i V2 slave	236
How to insert a slave device into an existing AS-i V2 configuration.	237
How to modify the address of an AS-i V2 device	238

Introduction to the Debug function

At a Glance

The function **Debugging** makes it possible for each AS-i V2 communication module present in the project:

- to display the slave state (connection, parameters etc.),
- to access the adjustment function for the selected channel (channel forcing etc.).

The function also accesses module diagnostics in the event of a fault.

Note: This function is only available in online mode.

Rack Display

It is also possible to access AS-i V2 module information from the **Rack display** page of a Web FactoryCast sever embedded in a **TSX ETY 4102** or **TSX ETY 5102** module. These Web pages are then accessible from an Internet navigator.
Refer to the Ethernet debugging (See Premium and Atrium using Unity Pro Manual, Ethernet network, Software installation for Ethernet communication) manual and the FactoryCast User Manual for more information.

Description of an AS-i V2 module's debug screen

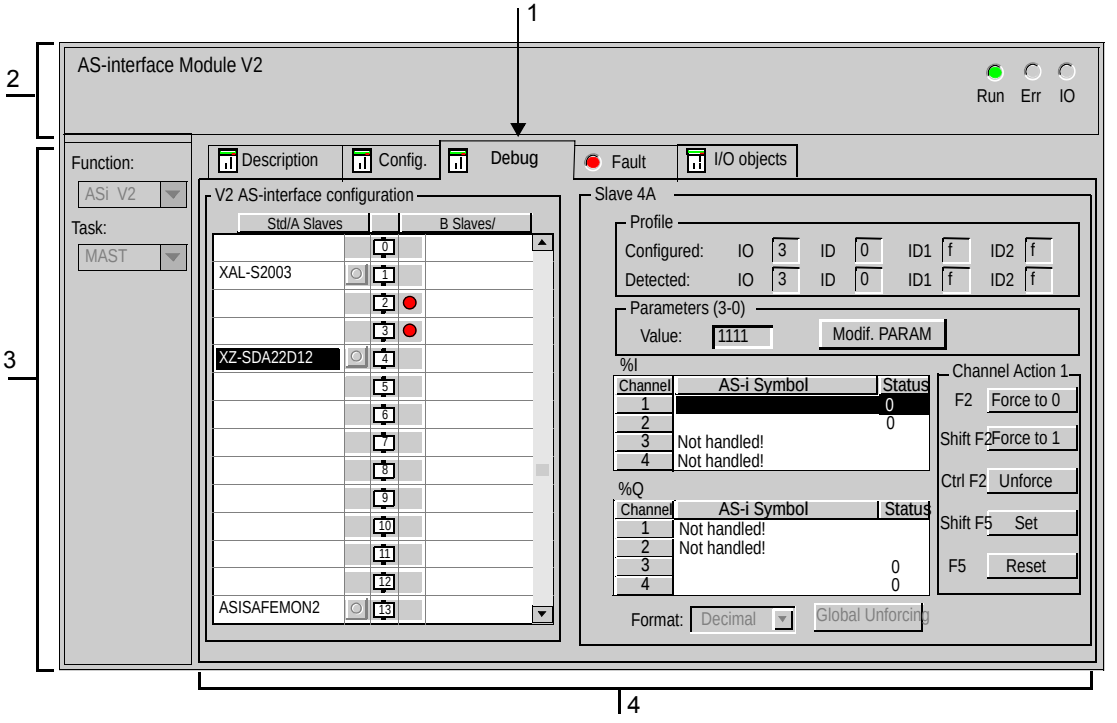
At a Glance

The debug screen dynamically displays the status of the AS-i V2 module and the devices which are connected to the bus.
It also allows access to the adjustment of the slave parameters and to the channel commands (forcing the input or output value, Set/Reset of an output,).

> **Note:** if in your user program you use the function `READ_STS ()` to read information relating to the AS-i V2 module, it is important not to carry out this function more than every 5s otherwise the debug screen will not display correctly.

Illustration

The figure below shows an example of a debug screen.



Description

The following table shows the different elements of the debug screen and their functions.

Address	Item	Function
1	Tabs	The front tab indicates the mode in progress (**Debug** for this example). Each mode can be selected with the corresponding tab. Possible modes are: • **Debug** only accessible only in online mode. • **Diagnostics** (Default), accessible only in Online mode, • **Configuration**. The tab **I/O Objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) is used to presymbolize the input/output objects.
2	**module** zone	Specify the abbreviated title of the module. In the same field there are 3 LEDs which provide information on the operating mode of the module • **RUN** indicates the operating mode of the module, • **ERR** signals an internal fault in the module, • **I/O** signals an external fault in the module or an application fault.
3	**General parameters** zone	Specify the parameterization of the task **MAST** or **FAST** configured for the AS-i communication channel.
4	**AS-i configuration and slave** area	Displays the slave devices connected to the bus. Displays the status of the slave's channels and gives access to the debug functions.

How to access module- and channel-diagnostics functions on an AS-i V2 device.

At a Glance

The functions of the module or channel diagnostics display the current errors in which are classed according to their category:

- internal errors (internal software errors, communication error with the processor, configuration, parameterization or command error),
- external faults (slave device failed, AS-i power supply switched off, terminal error, difference between physical configuration and Unity Pro configuration),
- other errors (module absent or switched off).

A faulty module and channel become apparent when certain lights change to red, such as:

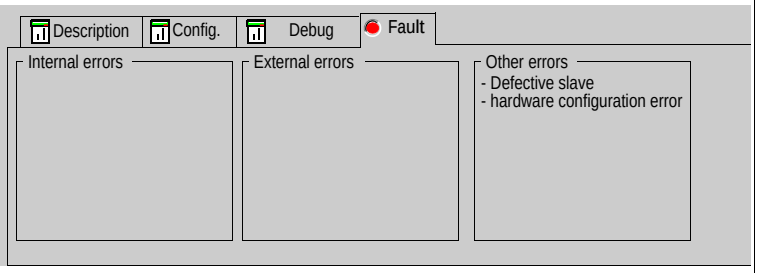
- in the rack configuration screen by the presence of a red square in the position of the faulty AS-i module,
- in all the module level screens (**Description** and **Default** tabs),
  - in the module zone with the LED **I/O**.
- in all the channel level screens (**Description**, **Config**, **Debug** and **Default** tabs),
  - in the module zone with the **I/O** LED,
  - in the channel zone with the LED of channel fault.
- in the faulty screen accessible with the **Fault** tab where the fault diagnostics are described.

The fault is also signaled:

- on the module, through the centralized display,
- by the dedicated language objects: **CH_ERROR** (%Ir.m.c.ERR) and module error **MOD_ERROR** (%Ir.m.MOD.ERR), **%MWr.m.MOD.2**, etc., and the status words (See *Details of the Language Objects of the IODDT of type T_GEN_MOD, p. 187*).

Procedure for accessing the module diagnostics

The following table presents the procedure to access the screen **Module diagnostics**.

Step	Action
1	Open the AS-i V2 module to be diagnosed.
2	Access the configuration screen by clicking on the **Fault** tab . **Result**: The list of module errors appears. 

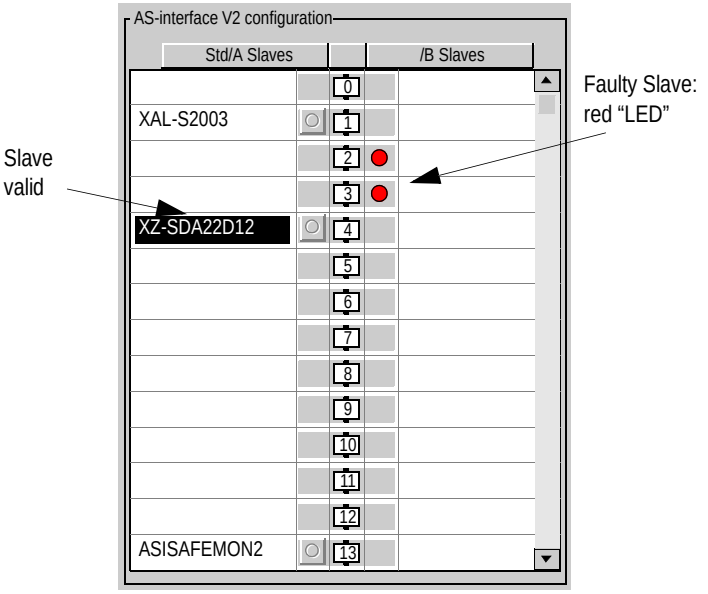
Displaying the slaves' status

At a Glance

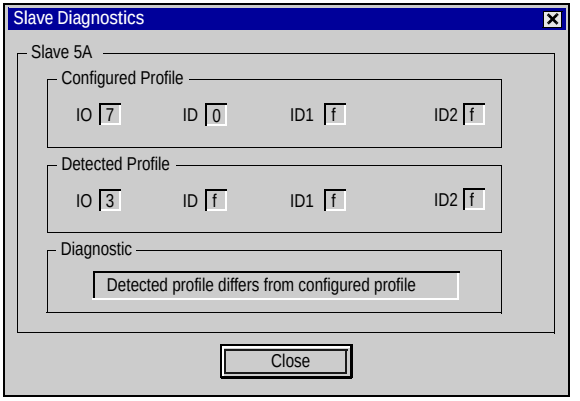
The lower part of the communication module's debug screen is reserved for AS-i V2 bus diagnostics.
The slave devices connected to the bus are displayed in the two columns of the **AS-interface V2 Configuration** zone. The left hand column lists standard (A) or extended (B) address slaves, and the right hand column lists only extended address (B) slaves only. An icon shows the status of the slave number.

Displaying slave status

Illustration



When you click on the slave button, a window shows the status of the slave, as shown in the diagram below:



The **Slave Diagnostics** window displays the following faults for each slave device:

- A configured profile is specified but no slave is detected.
- A profile is detected but there is no configured slave profile.
- A profile is detected but a different profile is specified in configuration (I/O, ID, ID1 or ID2).
- The peripheral fault, if the slave supports it.

Note: The **Profile** field in the **Slave Zone** in the debug screen allows you to check if the profiles of the specified (**Configured**) slave and the **Detected** slave are identical.

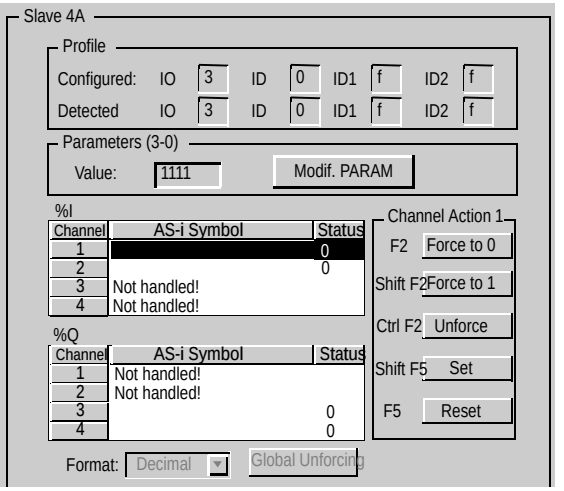
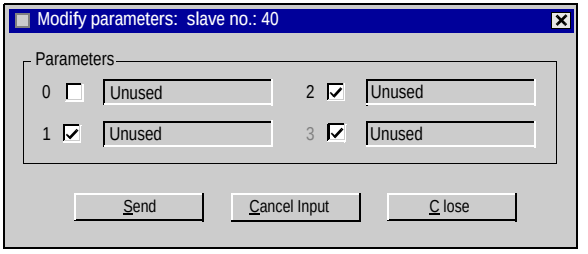
How to access the adjustment of the parameters of an AS-i V2 device

At a Glance

The debug screen of an AS-i V2 module allows the user, amongst other things, to modify a slave's parameters.

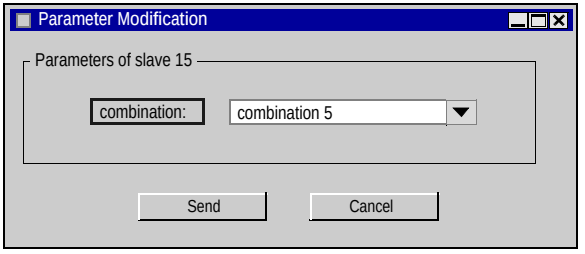
Procedure

The table below shows the procedure for modifying the parameters of a slave.

Step	Action
1	Access the AS-i V2 module's debug screen.
2	Select the slave in the **AS-interface V2 Configuration** zone. **Result**: In the slave zone of the debug screen, it is possible to read all the information relating to the selected slave. 
3	Click on **Modif. PARAM** located in the **Parameters** field of the slave zone. **Result**: The window**Parameter modification**appears. 
4	Selecting and omitting parameters.
5	Click on **Send** to recognize the new values.

Parameter Lists

Where a slave can support parameter lists, the procedure to follow is the same as above, but the window looks like this :



Select a combination of parameters from the drop-down list.

How to access the AS-i V2 digital channels' forcing/unforcing function

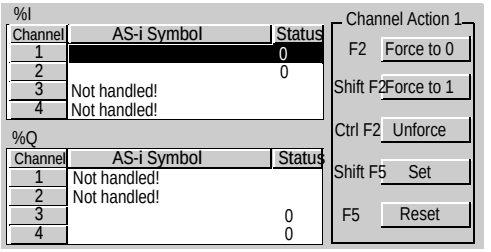
At a Glance

This function allows the status of channels linked to an AS-i V2 digital slave to be modified.
The different commands, which are available are:

- for a channel:
  - forcing to 0
  - forcing to 1
  - unforcing.
- for all the channels (when at least one channel is forced):
  - global unforcing.

Procedure

The table below shows the procedure to follow to force or unforce the channels linked to an AS-i V2 slave.

Step	Action for one channel	Action for a group of channels
1	Access the AS-i V2 module's debugging screen.	
2	Select a slave in the **AS-interface V2 Configuration** zone.	
3	Select the channel to modify in the slave zone table. **Result**: The channel can be modified using the buttons situated in the **Channel action** field. 	Click on the button **Global unforcing** found in the slave zone.
4	Select the required function (buttons **Force to 0** or **Force to 1** in the field **Channel Action**.	

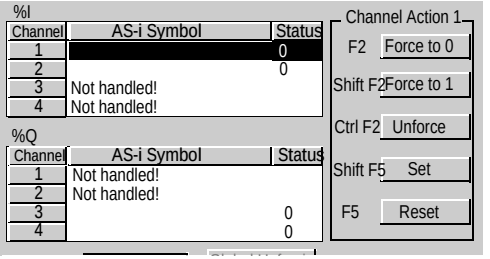
How to access the SET and RESET commands of the AS-i V2 digital channels

At a Glance

These commands allow values 0 (RESET) or 1 (SET) to be assigned to the channels of an AS-i V2 digital slave.
The status of the output affected by one of these commands is temporary and can be modified at any time by the project.

Procedure

The table below shows the procedure for assigning a value of 0 or 1 to the selected AS-i V2 slave's channels.

Step	Action
1	Access the AS-i V2 module's debugging screen.
2	Select a slave in the **AS-interface V2 Configuration** zone.
3	Select the channel to be modified in the **Slave** zone table. **Result**: The channel can be modified using the buttons situated in the **Channel action** field. 
4	Select the required function (buttons**Set** or **Reset**) in the field **Channel action**.

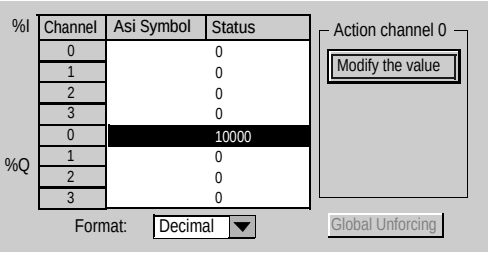
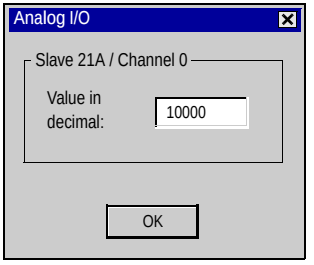
How to modify the value of an analog channel

At a Glance

This function allows the values of channels linked to an AS-i V2 analog slave to be modified.

Procedure

The following table shows the procedure for modifying the value of an AS-i V2 slave's analog channel.

Step	Action
1	Access the AS-i V2 module's debug screen.
2	Select an analog slave in the **AS configuration-V2 interface**zone.
3	Select the channel whose value you wish to modify from the **Slave** zone. **Result**: the following window appears : 
4	Click on the **Change Value** button. **Result**: the following window appears : 
5	Enter the value in decimals and click on **OK**. **Note**: the value is always entered as decimal, but it may be displayed in a different format using the **Format** drop-down list, located at the bottom of the **Slave** zone.

Automatic replacement of a faulty AS-i V2 slave

Principle

When a slave has been declared faulty, it can be automatically replaced with a slave of the same type.
The replacement happens without the AS-i V2 bus having to stop and without particular manipulation as long as the **Automatic addressing** (See *How to modify the general parameters of an AS-i V2 slave: Automatic addressing, p. 218*) utility of the configuration mode is active.

Two options are available:

- The replacement slave is programmed with the same address using the pocket programmer, and has the same profile and sub-profile as the faulty slave. It is thus automatically inserted into the list of detected slaves (LDS) and into the list of active slaves (LAS),
- The replacement slave is blank (address 0 (A), new slave) and has the same profile as the faulty slave. It will automatically assume the address of the replaced slave, and will then be inserted into the list of detected slaves (LDS) and the list of active slaves (LAS).

How to insert a slave device into an existing AS-i V2 configuration.

At a Glance

It is possible to insert a device into an existing AS-i V2 configuration without having to use the pocket programmer.
This operation is possible once:

- the configuration mode's **Automatic addressing** (See *How to modify the general parameters of an AS-i V2 slave: Automatic addressing, p. 218*) utility is active,
- a single slave is absent in the Unity Pro configuration,
- the slave which is to be inserted is specified in the -Unit configuration,
- the slave has the profile and sub-profile expected by the configuration,
- the slave has the address 0 (A).

The AS-i V2 module will therefore automatically assign to the slave the value predefined in the configuration.

Procedure

The following table shows the procedure for making the automatic insertion of a new slave effective.

Step	Action
1	Add the new slave in the configuration screen in local mode.
2	Carry out a configuration transfer to the PLC in connected mode.
3	Physically link the new slave with address 0 (A) to the AS -i V2 bus.

Note: A project can be modified by carrying out the above manipulation as many times as necessary.

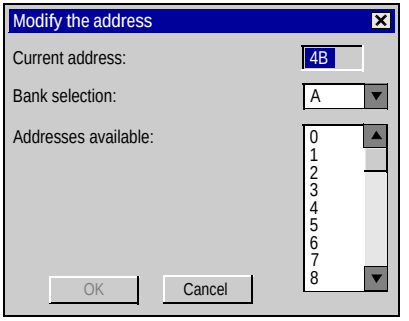
How to modify the address of an AS-i V2 device

At a Glance

This command allows the movement of the selected AS-V2 device to another available address. This address modification can only be done on the slave and not in configuration: thus physical configuration and software configuration are different.

Procedure

The following table shows the procedure for modifying the address of an AS-i V2 slave device.

Step	Action
1	Access the **TSX SAY 1000** module's debug screen.
2	Select a slave in the **AS-interface V2 configuration** then execute the command **Edit** → **Modify the address of an AS-i slave**. **Result**: The selection screen for a new address is displayed. 
3	Select bank A or B in the **Bank Selection** list. **Note**: bank B cannot be selected where there is a standard address slave.
4	Select the required address in the list of **Available addresses** (use the scroll bar if necessary).
5	Confirm the selection with **OK**.

AS-i V2 operating mode

15

At a Glance

Aim of this chapter

This chapter introduces the different AS-i V2 function operating modes.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
AS-i V2 Operating Mode: General	240
AS-i V2 protected Mode	242
AS-i V2 Offline Operating Mode	243
AS-i V2 Data Exchange Off Operating Mode	244

AS-i V2 Operating Mode: General

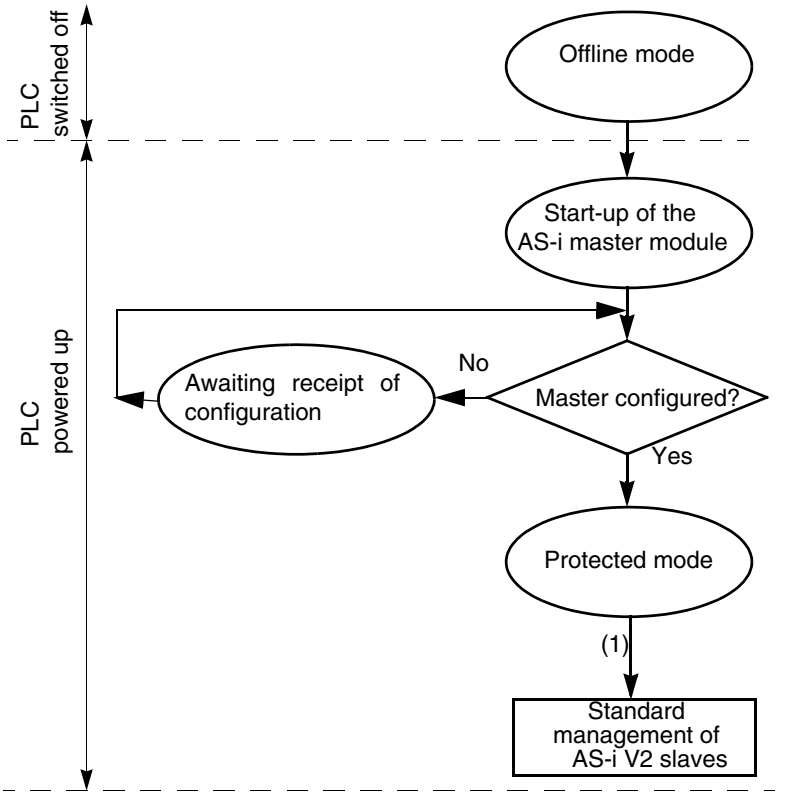
At a Glance

The extended AS-i V2 function allows three operating modes, each one fulfilling particular needs. These modes are:

- protected mode,
- offline mode,
- data exchange off mode.

Mode of Functioning

The figure below shows the general operating diagram of the AS-i V2



(1) : It is possible to toggle between the protected mode and the offline (See *AS-i V2 Offline Operating Mode, p. 243*) or data exchange off (See *AS-i V2 Data Exchange Off Operating Mode, p. 244*) modes.

Correspondence between PLC operating modes and the AS-i bus

The table below shows the correspondence between Premium PLC operating modes and those of the AS-i V2 bus.

PLC	AS-i Bus
configured mode (1)	Protected mode
non configured mode (1)	configuration mode
Legend:	
(1)	These PLC notions (configured, non-configured) concern the declaration of the module and slave devices in the Unity Pro project hardware configuration screen.

AS-i V2 protected Mode

At a glance

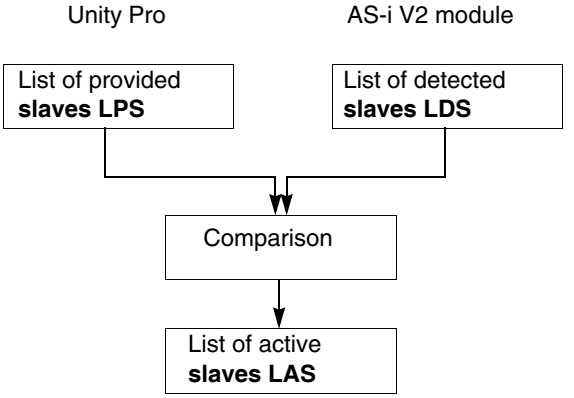
The AS-i V2 protégé operating mode is the mode generally used for a project which is running.
It assumes that the AS-i V2 module is configured in Unity Pro.
This:

- continually checks that the list of detected slaves is the same as the list of provided slaves,
- monitors the power supply.

In this mode, a slave will only be activated if it has been declared in the configuration and been detected.

Principle of activating a slave

The diagram below shows the principle of activating AS-i V2 slaves.



AS-i V2 Offline Operating Mode

At a Glance

The **Offline** AS-i V2 mode is an advanced operating mode which can be used in debugging or maintenance.

Note: The user must be thoroughly familiar with AS-i V2 communication.

Principle

When **Offline** mode is engaged, the module first resets all the present slaves to zero and stops exchanges on the bus.
During **Offline** mode, the outputs are forced to zero.

Note: The **Offline** mode is also used to exploit the infrared addressing function on the slaves which support this function.

Procedure

The **Offline** operating mode is accessible from the Unity Pro application software by acting on bits 0 and 1 of the word %MWr.m.0.74.

Reminder: Bit 7 of word %MWr.m.0.3 set on 1 indicates that the AS-i bus is in **Offline** mode.

AS-i V2 Data Exchange Off Operating Mode

At a glance

The **Data Exchange Off** AS-i V2 mode is an advanced operating mode which can be used in debugging or maintenance.

Note: The user must be thoroughly familiar with AS-i V2 communication.

Principle

When the **Data Exchange Off** mode is engaged, exchanges on the bus continue to function, but data is no longer refreshed.

Procedure

The **Data Exchange Off** operating mode can be accessed from the Unity Pro application software by acting on bits 2 and 3 of the word %MWr.m.c.74.

AS-i V2 performance

16

Performances of the AS-i V2 bus

Introduction

The AS-i V2 bus is independently managed by the master. This exchanges data on each cycle with each slave device configured on the bus (in ascending order of slave address number).

AS-i V2 Scanning Time

Scanning time **t** represents the exchange time between the master and **n** slaves (31 maximum on /A and/or /B).
So, for:

- up to 19 active slaves, t = 3ms
- from 20 to 31 active slaves t = (1 + number of active slaves) * 0.156 ms

When two slaves A and B have the same address, each slave in the pair is scanned every two cycles.
Thus, for 31 extended address slaves in /A + 31 extended address slaves in /B, the scanning time will be 10ms.

AS-i V2 Response Time

The response time **T** represents the AS-i V2 cycle time.
This includes:

- the bus scanning time,
- the update of the AS-i V2 module internal memory,
- the PLC cycle.

Example

The table below shows three examples of T response time for a PLC task lasting 10 ms, 30 ms, and 60 ms.
This T time is for a bus loaded with 31 slaves operating normally with no link faults.

PLC task	Typical response time	Maximum response time
10 ms	35 ms	56 ms
30 ms	65 ms	96 ms
60 ms	110 ms	156 ms

DFB for AS-i Security Monitor

17

At a Glance

Aim of this Section

This DFB describes the DFB for the AS-i Security Monitor.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Description of safety function block of the As-i bus	248
Operation of the AS-i security function block	251
Security DFB programming rules	252
How to implement a DFB function block	254

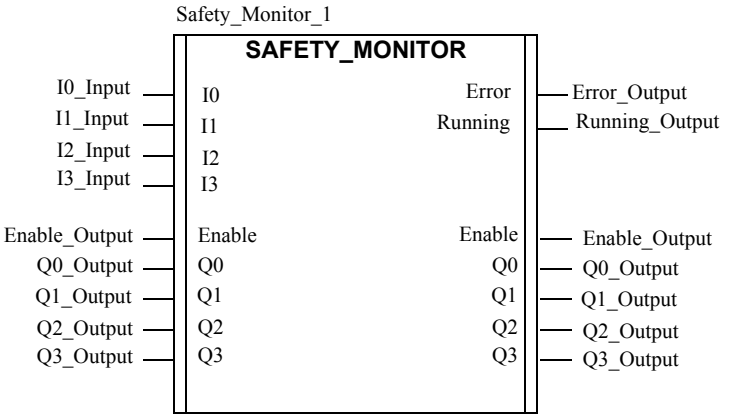
Description of safety function block of the As-i bus

Description of the function

This DFB allows data processed by the security monitor to be obtained.

Representation in FBD

Representation of the security DFB:



Description of the parameters

The following table describes the input parameters:

Parameter	Type	Description
I0	EBOOL	Input variable 0: %I\b.e\r.m.0.
I1	EBOOL	Input variable 1: %I\b.e\r.m.1.
I2	EBOOL	Input variable 2: %I\b.e\r.m.2.
I3	EBOOL	Input variable 3: %I\b.e\r.m.3.
Legend:		
b: AS-i bus number. e: slave number. r: rack. m: module.		

The following table describes the input/output parameters:

Parameter	Type	Description
Enable	EBOOL	Activation of the DFB (Cold Start): if this bit is at "1", the DFB executes, otherwise it is deactivated. The information is only usable if Enable = 0.
Q0	EBOOL	Output variable 0: %Q\b.e\r.m.0.
Q1	EBOOL	Output variable 1: %Q\b.e\r.m.1.
Q2	EBOOL	Output variable 2: %Q\b.e\r.m.2.
Q3	EBOOL	Output variable 3: %Q\b.e\r.m.3.
Legend:		
b: AS-i bus number. e: slave number. r: rack. m: module.		

The following table describes the output parameters:

Parameter	Type	Description
Error	EBOOL	This bit is set at "1" if an error appears: DFB error or security bus error (at least one slave is faulty): • if it is a DFB error (enable = 0): see Dfb_error (≠ 0) for more information. In the case of a DFB error, the information on the security project are no longer valid. • it is a case of a security project error (Dfb_error = 0 and Enable = 1): see S1_ to find out the faulty slaves.
Running	EBOOL	This bit is set at "1" during execution of the DFB.

The following table describes the internal public variables:

Name	Type	Description
Abort	EBOOL	If this bit is at "0" in one cycle and at "1" in the next, all the changes between CPU and the security monitor are stopped and the DFB reinitializes. All the internal data of the DFB are set at 0.
Timeout	INT	Timeout of the data exchanges (time base of 100 ms). If the DFB does not receive a correction before this period, the transaction is cancelled, the DFB is deactivated and the error bit is set at "1" (Dfb_stat and Dfb_err are updated).

Name	Type	Description
Moni_err	EBOOL	this bit is at "1" if the monitor is faulty.
Out_1	EBOOL	this bit is at "1" if the OUT1 contact is closed.
Out_2	EBOOL	this bit is at "1" if the OUT2 contact is closed.
SI_ready	DINT	each bit corresponds to the index of the security device that is in a test or read state.
SI_off	DINT	each bit corresponds to the index of the security device that is deactivated.
SI_error	DINT	each bit corresponds to the index of the security device that is faulty.
Dfb_stat	INT	this is the state of the DFB; this variable allows the user to check the progress of the DFB (See *Operation of the AS-i security function block, p. 251*).
Dfb_err	INT	This word gives the error type: ● 16#90 ; the reply sent by the monitor is invalid, ● 16#91 : the DFB has been deleted by the user, ● 16#92 : the exchange has stopped on a Timeout, the DFB cannot receive data.

Searching for a security slave that has been activated

The search for the address of a security slave that has been activated can be facilitated by taking into account inputs I3/I2/I1/I0 of the slave. These inputs are equal to 0/0/0/0 if the slave is activated, and different from 0/0/0/0 if the slave is not activated. This information must not be taken into account in the security: only the security monitor provides security outputs.

Operation of the AS-i security function block

General operation

All the information used in the AS- security DFB is obtained from language objects associated to the AS-i TSX SAY 100 and AS-i V2 TSX SAY 1000 modules.

Step by step description of operation

The table below describes, step by step, the operation of the security DFB:

Step	Action
1	The bus master sends the request to test the monitor.
2	The bus master sends the request for the monitor to copy its state to the static memory.
3	The bus master analyses the data from the monitor.
4	Recovery of the data of all the security devices.

Security DFB programming rules

General

The security DFB, the implementation of which is identical to that of a DIAG AS-i DFB, can be programmed in any program module (Main, SR or section) in Ladder Diagram (LD), Structured Text (ST), Functional Block (DFB) and Instruction List (IL) languages.
It is:

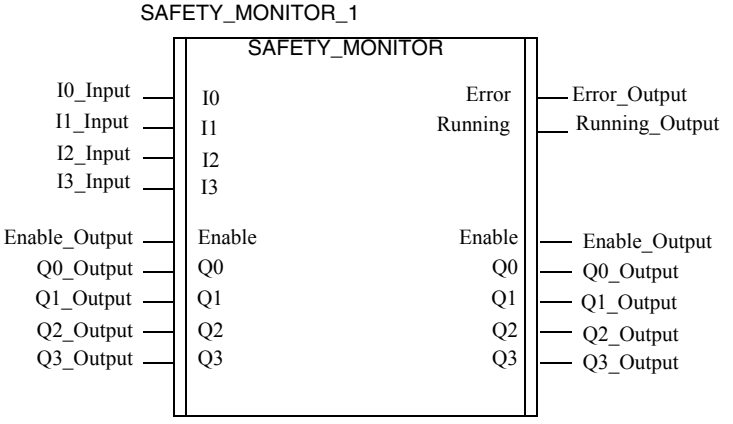
- available systematically in Unity Pro (See *, p. 255*),
- read and write protected,
- dedicated to a single security monitor.

Rules

- It is preferable to run the security DFB in the MAST task (for performance reasons).
- It is strongly recommended to program a security DFB instance only once in the project.
- In order for a security DFB to execute, it is necessary that:
  - the DFB is called (the program element to which it is assigned must execute),
  - the `Enable` input is at 1.

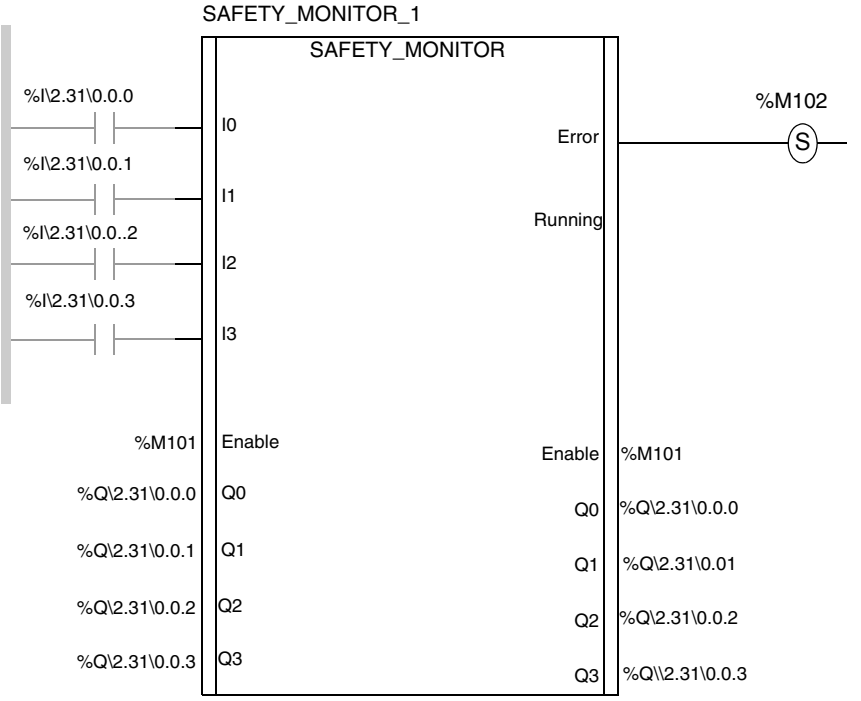
Programming in FBD

Representation:



Programming in ladder diagram

The security DFB function block is inserted in a contacts network. The programming consists of linking these inputs/outputs



Representation in IL

Representation:

```
CAL SAFETY_MONITOR_1 (I0,..., I3, %MW101, Q0,...,Q3, %M102)
```

Programming in structured text language

The programming syntax is as follows:

```
%Li: label
SAFETY_MONITOR_1 ( I0,..., I3, %MW101, Q0,...,Q3, %M102);
```

where:

- `%Li`: label,
- `SAFETY_MONITOR_1`: name of instance of a DFB,
- `I0,..., I3`: DFB inputs,
- %M101: DFB enable input,
- `Q0,...,Q3`: variable linked with the DFB outputs,
- %M102: Error output variable

How to implement a DFB function block

Reminder

The AS-i security DFB exists in the Diagnostics Library.

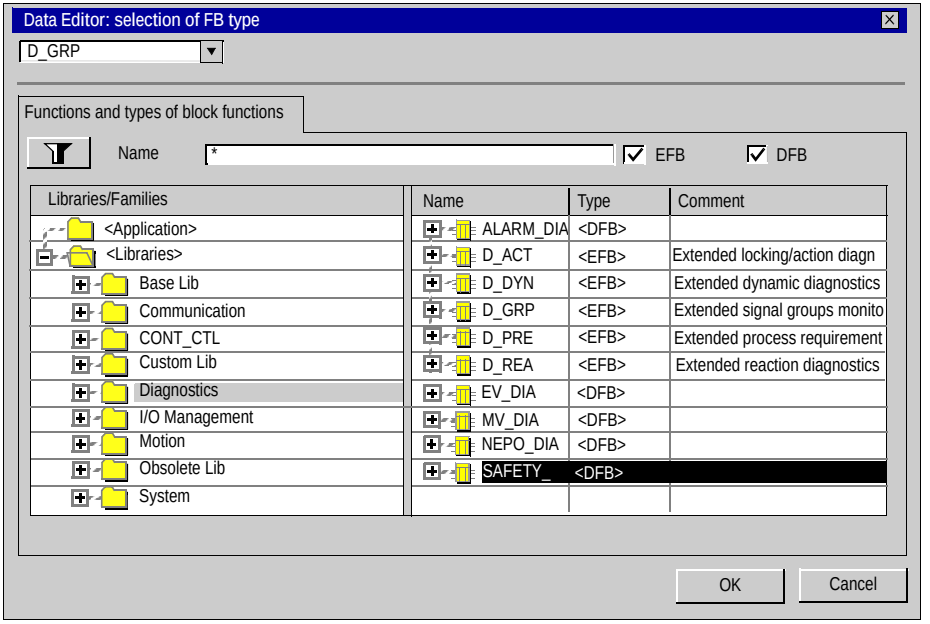
Use of DFBs

This instance of the block is used as a standard function block in ladder diagram, or as an elementary function in structured literal, instruction list or function block language.

It can be programmed in the various tasks (except in the event tasks) and sections of the project.

Security DFB instance

The following table describes the procedure for creating a security DFB instance:

Step	Action
1	Select the **Function blocks** tab in the data editor window.
2	Check the box marked **DFB**.
3	Declare a DFB instance in the **Name** column.
4	To select the DFB type double click in the **Type** column of the new instance and press on the button . **Result**: the **Data editor window: selection of FB type** opens.
5	Select the **Libraries** directory in the **<Libraries/Families>** zone.
6	In the **<Libraries>** zone select the **Diagnostics** directory. **Result**: the various diagnostics DFBs appear in the right hand side of the screen: 
7	Select the **SAFETY_MONITOR** DFB type in the **Name** column and press **OK** to confirm.
8	Program (See *Security DFB programming rules, p. 252*) the DFB in the desired program module.

Language objects of the AS-i V2 bus

18

At a glance

Aim of this Chapter

This chapter describes the language objects associated to the AS-i V2 bus as well as the different ways of using them.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
18.1	Language objects and IODDT's of the AS-i V2 communication	258
18.2	Language objects and generic IODDT for all the communication protocols	259
18.3	Language objects of the AS-i V2 function	263
18.4	IODDT T_GEN_MOD applicable to all the modules	279

18.1 Language objects and IODDT's of the AS-i V2 communication

Presentation of language objects of the AS-i V2 bus

General

The IODDT's are predefined by the constructor and contains language objects of inputs/outputs belonging to the channel of an application module.
AS-i V2 communication has two associated IODDT's:
- `T_COM_STS_GEN` which applies to all the communications protocols,
- `T_COM_ASI_V2` specific to AS-i V2 communication and the TSX SAY 1000 module.

Note: IODDT variables can be created in two different ways:
- Using the **I/O objects** (See Unity Pro, Operating Modes Manual, I/O Objects Tab for a Module) tab,
- Data Editor (See Unity Pro, Operating Modes Manual, Creation of an IODDT type data instance).

Language object types

Each IODDT contains a set of language objects allowing its operation to be controlled and checked.

There are two types of language objects:
- **implicit exchange objects**, (See *Implicit Exchange Language Objects Associated with the Application-Specific Function, p. 162*) which are automatically exchanged at each cycle of the task associated with the module,
- **explicit exchange objects** (See *Explicit exchange language objects associated with the application-specific function, p. 163*), which are exchanged when requested to do so by the project, using explicit exchange instructions.

Note: when using explicit exchange objects with the READ_STS() function it is important not to carry out this function more than every 5s, otherwise the debug screen of the module TSX SAY 1000 will not display in Online mode.

Implicit exchanges concern the states of the modules, communication signals, slaves, and so on.
Explicit exchanges allow module parametering and diagnostics.

18.2 Language objects and generic IODDT for all the communication protocols

At a Glance

Object of this sub-chapter

This sub-chapter presents the language objects and the generic IODDT that apply to all the communication protocols.

What's in this Section?

This section contains the following topics:

Topic	Page
Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type	260
Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type	261

Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type

At a Glance

The following table presents the implicit exchange objects of the IODDT of the `T_COM_STS_GEN` type that applies to all the communication protocols.

Error bit

The following table shows the meaning of the error bit`CH_ERROR` (%Ir.m.c.ERR).

Standard symbol	Type	Access	Meaning	Address
CH_ERROR	EBOOL	R	Communication channel error bit	%Ir.m.c.ERR

Details of the implicit exchange objects of the IODDT of the T_COM_STS_GEN type

At a Glance

This part presents the explicit exchange objects of the IODDT of the T_COM_STS_GEN type that applies to all the communication protocols. This groups together word type objects, the bits of which have a particular meaning. These objects are presented in detail below
Example of declaration of a: **IODDT_VAR1** variable of type T_COM_STS_GEN.

Notes

- Generally the meaning of the bits is given for state 1 of each bit. In certain cases each state of the bit is explained.
- Not all the bits are used.

Explicit exchange indicator: EXCH_STS

The following table presents the meanings of the exchange control bits of the channel EXCH_STS (%MWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading of the status words of the current channel.	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	R	Command parameters exchange in progress.	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Command parameters adjustment in progress.	%MWr.m.c.0.2

Explicit exchange report: EXCH_RPT

The table below presents the meanings of the report bitsEXCH_RPT (%MWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Channel status words reading error.	%MWr.m.c.1.0
CMD_ERR	BOOL	R	Fault during a command parameters exchange.	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	Fault during an adjustment parameters exchange.	%MWr.m.c.1.2

Standard channels faults, CH_FLT

The table below presents the meanings of the status word bits CH_FLT (%MWr.m.c.2). The reading is performed by a **READ_STS**(**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
NO_DEVICE	BOOL	R	No device is working on the channel.	%MWr.m.c.2.0
1_DEVICE_FLT	BOOL	R	A device on the channel is faulty.	%MWr.m.c.2.1
BLK	BOOL	R	Terminal strip error (not connected).	%MWr.m.c.2.2
TO_ERR	BOOL	R	Timeout error (faulty wiring).	%MWr.m.c.2.3
INTERNAL_FLT	BOOL	R	Internal fault or channel self-test.	%MWr.m.c.2.4
CONF_FLT	BOOL	R	Different hardware and software configurations.	%MWr.m.c.2.5
COM_FLT	BOOL	R	Problem communicating with the PLC.	%MWr.m.c.2.6
APPLI_FLT	BOOL	R	Application fault (adjustment or configuration fault).	%MWr.m.c.2.7

18.3 Language objects of the AS-i V2 function

At a Glance

Aim of this Section

This section provides general information on the language objects of the AS-i V2 function and its associated IODDT.

What's in this Section?

This section contains the following topics:

Topic	Page
Details of the implicit exchange objects of T_COM_ASI_V2 type IODDTs	264
Details of the explicit exchange objects of the T_COM_ASI_V2 type IODDT	269
Details of implicit exchange objects of the AS-i V2 function	271
Details of explicit exchange objects of the AS-i V2 function	274
Details of the objects managing the mode of operation for the AS-i V2 function	277
Details of configuration objects for the AS-i V2 function	278

Details of the implicit exchange objects of T_COM_ASI_V2 type IODDTs

At a Glance

The tables below present the implicit exchange objects of `T_COM_ASI_V2` type IODDTs that apply to AS-i V2 communication.

Error bit

The following table shows the meaning of the `CH_ERROR` (%Ir.m.c.ERR) error bit.

Standard symbol	Type	Access	Meaning	Address
CH_ERROR	BOOL	R	Communication channel error bit.	%Ir.m.c.ERR

Validity bit

The following table shows the meaning of the `VALID_IN` (%Ir.m.c.0) validity bit.

Standard symbol	Type	Access	Meaning	Address
VALID_IN	EBOOL	R	Indicates that all inputs are valid. Note: when this bit is on 0, it indicates that at least one input is invalid: offline, data exchange off, or channel error mode.	%Ir.m.c.0

List of slaves 0 to 15 of bank A in error

The following table shows the meaning of the bits of word (%IWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_0A	BOOL	R	Slave 0A faulty or absent.	%IWr.m.c.0.0
SLAVE_FLT_1A	BOOL	R	Slave 1A faulty or absent.	%IWr.m.c.0.1
SLAVE_FLT_2A	BOOL	R	Slave 2A faulty or absent.	%IWr.m.c.0.2
SLAVE_FLT_3A	BOOL	R	Slave 3A faulty or absent.	%IWr.m.c.0.3
SLAVE_FLT_4A	BOOL	R	Slave 4A faulty or absent.	%IWr.m.c.0.4
SLAVE_FLT_5A	BOOL	R	Slave 5A faulty or absent.	%IWr.m.c.0.5
SLAVE_FLT_6A	BOOL	R	Slave 6A faulty or absent.	%IWr.m.c.0.6
SLAVE_FLT_7A	BOOL	R	Slave 7A faulty or absent.	%IWr.m.c.0.7
SLAVE_FLT_8A	BOOL	R	Slave 8A faulty or absent.	%IWr.m.c.0.8
SLAVE_FLT_9A	BOOL	R	Slave 9A faulty or absent.	%IWr.m.c.0.9
SLAVE_FLT_10A	BOOL	R	Slave 10A faulty or absent.	%IWr.m.c.0.10
SLAVE_FLT_11A	BOOL	R	Slave 11A faulty or absent.	%IWr.m.c.0.11
SLAVE_FLT_12A	BOOL	R	Slave 12A faulty or absent.	%IWr.m.c.0.12
SLAVE_FLT_13A	BOOL	R	Slave 13A faulty or absent.	%IWr.m.c.0.13
SLAVE_FLT_14A	BOOL	R	Slave 14A faulty or absent.	%IWr.m.c.0.14
SLAVE_FLT_15A	BOOL	R	Slave 15A faulty or absent.	%IWr.m.c.0.15

List of slaves 16 to 31 of bank A in error

The following table shows the meaning of the bits of word (%IWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_16A	BOOL	R	Slave 16A faulty or absent.	%IWr.m.c.1.0
SLAVE_FLT_17A	BOOL	R	Slave 17A faulty or absent.	%IWr.m.c.1.1
SLAVE_FLT_18A	BOOL	R	Slave 18A faulty or absent.	%IWr.m.c.1.2
SLAVE_FLT_19A	BOOL	R	Slave 19A faulty or absent.	%IWr.m.c.1.3
SLAVE_FLT_20A	BOOL	R	Slave 20A faulty or absent.	%IWr.m.c.1.4
SLAVE_FLT_21A	BOOL	R	Slave 21A faulty or absent.	%IWr.m.c.1.5
SLAVE_FLT_22A	BOOL	R	Slave 22A faulty or absent.	%IWr.m.c.1.6
SLAVE_FLT_23A	BOOL	R	Slave 23A faulty or absent.	%IWr.m.c.1.7
SLAVE_FLT_24A	BOOL	R	Slave 24A faulty or absent.	%IWr.m.c.1.8
SLAVE_FLT_25A	BOOL	R	Slave 25A faulty or absent.	%IWr.m.c.1.9
SLAVE_FLT_26A	BOOL	R	Slave 26A faulty or absent.	%IWr.m.c.1.10
SLAVE_FLT_27A	BOOL	R	Slave 27A faulty or absent.	%IWr.m.c.1.11
SLAVE_FLT_28A	BOOL	R	Slave 28A faulty or absent.	%IWr.m.c.1.12
SLAVE_FLT_29A	BOOL	R	Slave 29A faulty or absent.	%IWr.m.c.1.13
SLAVE_FLT_30A	BOOL	R	Slave 30A faulty or absent.	%IWr.m.c.1.14
SLAVE_FLT_31A	BOOL	R	Slave 31A faulty or absent.	%IWr.m.c.1.15

List of slaves 0 to 15 of bank B in error

The following table shows the meaning of the bits of the word (%IWr.m.c.2).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_0B	BOOL	R	Slave 0B faulty or absent.	%IWr.m.c.2.0
SLAVE_FLT_1B	BOOL	R	Slave 1B faulty or absent.	%IWr.m.c.2.1
SLAVE_FLT_2B	BOOL	R	Slave 2B faulty or absent.	%IWr.m.c.2.2
SLAVE_FLT_3B	BOOL	R	Slave 3B faulty or absent.	%IWr.m.c.2.3
SLAVE_FLT_4B	BOOL	R	Slave 4B faulty or absent.	%IWr.m.c.2.4
SLAVE_FLT_5B	BOOL	R	Slave 5B faulty or absent.	%IWr.m.c.2.5
SLAVE_FLT_6B	BOOL	R	Slave 6B faulty or absent.	%IWr.m.c.2.6
SLAVE_FLT_7B	BOOL	R	Slave 7B faulty or absent.	%IWr.m.c.2.7
SLAVE_FLT_8B	BOOL	R	Slave 8B faulty or absent.	%IWr.m.c.2.8
SLAVE_FLT_9B	BOOL	R	Slave 9B faulty or absent.	%IWr.m.c.2.9
SLAVE_FLT_10B	BOOL	R	Slave 10B faulty or absent.	%IWr.m.c.2.10
SLAVE_FLT_11B	BOOL	R	Slave 11B faulty or absent.	%IWr.m.c.2.11
SLAVE_FLT_12B	BOOL	R	Slave 12B faulty or absent.	%IWr.m.c.2.12
SLAVE_FLT_13B	BOOL	R	Slave 13B faulty or absent.	%IWr.m.c.2.13
SLAVE_FLT_14B	BOOL	R	Slave 14B faulty or absent.	%IWr.m.c.2.14
SLAVE_FLT_15B	BOOL	R	Slave 15B faulty or absent.	%IWr.m.c.2.15

List of slaves 16 to 31 of bank B in error

The following table shows the meaning of the bits of the word (%IWr.m.c.3).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT_16B	BOOL	R	Slave 16B faulty or absent.	%IWr.m.c.3.0
SLAVE_FLT_17B	BOOL	R	Slave 17B faulty or absent.	%IWr.m.c.3.1
SLAVE_FLT_18B	BOOL	R	Slave 18B faulty or absent.	%IWr.m.c.3.2
SLAVE_FLT_19B	BOOL	R	Slave 19B faulty or absent.	%IWr.m.c.3.3
SLAVE_FLT_20B	BOOL	R	Slave 20B faulty or absent.	%IWr.m.c.3.4
SLAVE_FLT_21B	BOOL	R	Slave 21B faulty or absent.	%IWr.m.c.3.5
SLAVE_FLT_22B	BOOL	R	Slave 22B faulty or absent.	%IWr.m.c.3.6
SLAVE_FLT_23B	BOOL	R	Slave 23B faulty or absent.	%IWr.m.c.3.7
SLAVE_FLT_24B	BOOL	R	Slave 24B faulty or absent.	%IWr.m.c.3.8
SLAVE_FLT_25B	BOOL	R	Slave 25B faulty or absent.	%IWr.m.c.3.9
SLAVE_FLT_26B	BOOL	R	Slave 26B faulty or absent.	%IWr.m.c.3.10
SLAVE_FLT_27B	BOOL	R	Slave 27B faulty or absent.	%IWr.m.c.3.11
SLAVE_FLT_28B	BOOL	R	Slave 28B faulty or absent.	%IWr.m.c.3.12
SLAVE_FLT_29B	BOOL	R	Slave 29B faulty or absent.	%IWr.m.c.3.13
SLAVE_FLT_30B	BOOL	R	Slave 30B faulty or absent.	%IWr.m.c.3.14
SLAVE_FLT_31B	BOOL	R	Slave 31B faulty or absent.	%IWr.m.c.3.15

Details of the explicit exchange objects of the T_COM_ASI_V2 type IODDT

At a Glance

This part introduces the explicit exchange objects of `T_COM_ASI_V2` type IODDTs that apply to the AS-i V2 bus. It concerns those word type objects whose bits have a particular meaning. These objects are presented below.
Example of declaration of a variable: **IODDT_VAR1** of type `T_COM_ASI_V2`

Notes

- The meaning of a bit is generally given for the status of the bit when set to 1. In specific cases an explanation is given for each status of the bit.
- Not all bits are used.

Execution indicators for an explicit exchange: EXCH_STS

The table below shows the meaning of channel exchange control bits `EXCH_STS` (%MWr.m.c.0).

Standard symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading current channel status words.	%MWr.m.c.0.0
CMD_IN_PROGR	BOOL	R	Command parameters exchange in progress.	%MWr.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Adjust parameters exchange in progress.	%MWr.m.c.0.2

Explicit exchange report: EXCH_RPT

The table below shows the meaning of the exchange report bits `EXCH_RPT` (%MWr.m.c.1).

Standard symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Channel status words reading error.	%MWr.m.c.1.0
CMD_ERR	BOOL	R	Command parameter exchange error.	%MWr.m.c.1.1
ADJ_ERR	BOOL	R	Adjust parameter exchange error.	%MWr.m.c.1.2

Standard channel faults, CH_FLT

The table below shows the meanings of the bits of the CH_FLT (%MWr.m.c.2) status word. Reading can be done via a READ_STS (**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
SLAVE_FLT	BOOL	R	One or more device faulty.	%MWr.m.c.2.1
ASI_CONF_FLT	BOOL	R	Physical configuration different from software configuration.	%MWr.m.c.2.3
INTERNAL_FLT	BOOL	R	Internal error or channel self-test.	%MWr.m.c.2.4
CONF_FLT	BOOL	R	Hardware or software configuration fault.	%MWr.m.c.2.5
COM_FLT	BOOL	R	Problem communicating with the PLC.	%MWr.m.c.2.6
APPLI_FLT	BOOL	R	Application fault (adjustment or configuration fault).	%MWr.m.c.2.7

Specific status AS-i V2

The table below presents the meanings of the (%MWr.m.c.3) status word bits (%MWr.m.c.2). Reading can be done via a READ_STS(**IODDT_VAR1**).

Standard symbol	Type	Access	Meaning	Address
ASI_SUPPLY_FLT	BOOL	R	AS-i power supply fault.	%MWr.m.c.3.6

Details of implicit exchange objects of the AS-i V2 function

At a Glance

The tables below present the different implicit exchange word objects. These word objects **are not** built into `T_COM_ASI_V2` type IODDTs.

Discrete input channel bits

The table below shows the meaning of the Discrete input channel bits for bank A (slaves 1 to 31) and bank B (slaves 101 to 131).

Address	Type	Access	Meaning
%I\b.e\r.m.c	EBOOL	R	Indicates that the input channel (c = 0 to 3) of slave number e is activated. For bank A: e =1 to 31. For bank B: e =101 to 131.
Examples:			
%I\b.0\r.m.c	EBOOL	R	Indicates that the input channel (c = 0 to 3) of slave number 0 is activated.
%I\b.1\r.m.c	EBOOL	R	Indicates that the input channel (c = 1 to 3) of slave number 0 is activated.
%I\b.110\r.m.c	EBOOL	R	Indicates that the input channel (c = 110 to 3) of slave number 0 is activated.
%I\b.131\r.m.c	EBOOL	R	Indicates that the input channel (c = 131 to 3) of slave number 0 is activated.

Discrete output channel bits

The table below shows the meaning of the Discrete output channel bits for bank A (slaves 1 to 31) and bank B (slaves 101 to 131).

Address	Type	Access	Meaning
%Q\b.e\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number e is activated. For bank A: e =1 to 31. For bank B: e =101 to 131.
Examples:			
%Q\b.0\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 0 is activated.
%Q\b.1\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 1 is activated.
%Q\b.110\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 110 is activated.
%Q\b.131\r.m.c	EBOOL	R	Indicates that the output channel (c = 0 to 3) of slave number 131 is activated.

Analog input channel words

The table below shows the meaning of the analog channel input words for bank A (slaves 1 to 31) and bank B (slaves 101 to 131).

Address	Type	Access	Meaning
%IW\b.e\r.m.c	INT	R	Value of the analog channel (c = 0 to 3) input word of slave number e For bank A: e =1 to 31. For bank B: e =101 to 131.

Analog output channel words

The table below shows the meaning of the analog channel output words for bank A (slaves 1 to 31) and bank B (slaves 101 to 131).

Address	Type	Access	Meaning
%QW\b.e\r.m.c	INT	R	Value of the analog channel (c = 0 to 3) output word of slave number e For bank A: e =1 to 31. For bank B: e =101 to 131.

Word objects

The table below shows the meaning of the different implicit exchange word objects.

Address	Type	Access	Meaning
%IWr.m.c.4.n	BOOL	R	n = 0 to 15: standard address slave (A) 0 to 15 (LAS), respectively, with peripheral fault (LPF).
%IWr.m.c.5.n	BOOL	R	n = 0 to 15: standard address slave (A) 16 to 31 (LAS), respectively, with peripheral fault (LPF).
%IWr.m.c.6.n	BOOL	R	n = 0 to 15: extended address slave (B) 0 to 15 (LAS), respectively, with peripheral fault (LPF).
%IWr.m.c.7.n	BOOL	R	n = 0 to 15: extended address slave (B) 16 to 31 (LAS), respectively, with peripheral fault (LPF).
%IWr.m.c.8.n	BOOL	R	n = 0 to 15: standard address slave (A) 0 to 15 (LAS), respectively, active.
%IWr.m.c.9.n	BOOL	R	n = 0 to 15: standard address slave (A) 16 to 31 (LAS), respectively, active.
%IWr.m.c.10.n	BOOL	R	n = 0 to 15: extended address slave (B) 0 to 15 (LAS), respectively, active.
%IWr.m.c.11.n	BOOL	R	n = 0 to 15: extended address slave (B) 16 to 31 (LAS), respectively, active.

Details of explicit exchange objects of the AS-i V2 function

At a Glance

The tables below present the different explicit exchange word objects. These word objects **are not** built into `T_COM_ASI_V2` type IODDTs.

Exchange control bit

The following table shows the meaning of the channel exchange control bit (%MWr.m.c.0).

Address	Type	Access	Meaning
%MWr.m.c.0.15	BOOL	R	Configuration in progress.

Exchange report bit

The table below shows the meaning of the report bit (%MWr.m.c.1).

Address	Type	Access	Meaning
%MWr.m.c.1.15	BOOL	R	Error in last configuration action

Mots %MWr.m.c.3

The table below shows the meaning of the bits of words %MWr.m.c.3.

Address	Type	Access	Meaning
%MWr.m.c.3.0	BOOL	R	Correct configuration.
%MWr.m.c.3.1	BOOL	R	Slave 0 present on the bus.
%MWr.m.c.3.2	BOOL	R	Automatic addressing.
%MWr.m.c.3.3	BOOL	R	Automatic addressing authorized.
%MWr.m.c.3.4	BOOL	R	Operation in AS-i configuration mode.
%MWr.m.c.3.6	BOOL	R	Operation in normal mode.
%MWr.m.c.3.6	BOOL	R	Faulty AS-i power supply.
%MWr.m.c.3.7	BOOL	R	Offline mode active.
%MWr.m.c.3.8	BOOL	R	Data exchange inactive.
%MWr.m.c.3.9	BOOL	R	Peripheral fault on a bus device.

Words %MWr.m.c.0.4 to %MWr.m.c.0.7

The table below shows the meaning of the bits of words %MWr.m.c.4 to %MWr.m.c.7.

Address	Type	Access	Meaning
%MWr.m.c.4.n	BOOL	R	n = 0 to 15 -> slaves of bank A from 0 to 15, respectively, detected.
%MWr.m.c.5.n	BOOL	R	n = 0 to 15 -> slaves of bank A from 16 to 31, respectively, detected.
%MWr.m.c.6.n	BOOL	R	n = 0 to 15 -> slaves of bank B from 0 to 15, respectively, detected.
%MWr.m.c.7.n	BOOL	R	n = 0 to 15 -> slaves of bank B from 16 to 31, respectively, detected.

Words %MWr.m.c.0.8 to %MWr.m.c.0.71

The table below shows the meaning of the bits of words %MWr.m.c.8 to %MWr.m.c.71..

Address	Type	Access	Meaning
%MWr.m.c.8 to %MWr.m.c.39	BOOL	R/W	Words 8 to 39 -> of the slaves of **bank A** from 0 to 31, respectively. Byte 0: bit 0 to 3 = configuration code for input/output (I/O) channels, bit 4 to 7 = identification code (ID), Byte 1: bit 0 to 3 = identification code (ID1), bit 4 to 7 = identification code (ID2).
%MWr.m.c.40 to %MWr.m.c.71	BOOL	R/W	Words 8 to 39 -> of the slaves of **bank B** from 0 to 31, respectively. Byte 0: bit 0 to 3 = configuration code for input/output (I/O) channels, bit 4 to 7 = identification code (ID), Byte 1: bit 0 to 3 = identification code (ID1), bit 4 to 7 = identification code (ID2).

%MWr.m.c.0.72 word

The table below gives the meaning of the word %MWr.m.c.72.

Address	Type	Access	Meaning
%MWr.m.c.72	INT	R	Contains the answer (value of parameters sent) of the last parametered slave. This is so that the Unity Pro can check the slave has correctly received these values.

%MWr.m.c.0.73 word

The table below gives the meaning of the word %MWr.m.c.73.

Address	Type	Access	Meaning
%MWr.m.c.73	INT	R	Contains the address of last parametered slave. Byte 0: Address of the slave (1 to 31), Byte 1: Slave bank: 0 = bank A, 1 = bank B.

Details of the objects managing the mode of operation for the AS-i V2 function

At a Glance

The tables below present the language objects allowing management of the operating mode of the AS-i V2 function and the parameters of the slaves present on the AS-i V2 bus. These objects are not built into the IODDT's.

Operating mode object

This word type object is used to manage the switch to Offline mode (See *AS-i V2 Offline Operating Mode, p. 243*) or Data Exchange Off mode (See *AS-i V2 Data Exchange Off Operating Mode, p. 244*) of the AS-i master.

Note: Use of this object requires a thorough knowledge of AS-i V2 communication principles.

Description

The table below presents the different states of the %MWr.m.c.74 word, giving access.

Address	Type	Access	Meaning
%MWr.m.c.74.0	BOOL	R/W	Switching to Offline mode
%MWr.m.c.74.1	BOOL	R/W	Exit from offline mode
%MWr.m.c.74.2	BOOL	R/W	Data exchange inactive.
%MWr.m.c.74.3	BOOL	R/W	Data exchange active.

Note: Offline mode has priority over Data Exchange Off.

Parameter objects

These objects are used to manage the parameters of slave devices. They can be modified without stopping the AS-i V2 function.

Description

The table below provides the meaning of the different words %MWr.m.c.75 to %MWr.m.c.138.

Address	Type	Access	Meaning
%MWr.m.c.75 to %MWr.m.c.106	INT	R/W	Contain the parameter values of slaves 0 to 31 respectively of **bank A**.
%MWr.m.c.107 to %MWr.m.c.138	INT	R/W	Contain the parameter values of slaves 0 to 31 respectively of **bank B**.

Details of configuration objects for the AS-i V2 function

At a Glance

%KWr.m.c.d. configuration constants are accessible in read-only mode and correspond to the configuration parameters entered using the Configuration editor.

Configuration objects

The table below shows constant objects on the AS-i V2 extended bus.

Address	Type	Access	Meaning
%KWr.m.c.0	INT	R	When byte 0 = 10, this shows that AS-i is master.
%KWr.m.c.1.n	BOOL	R	n = 0 to `15 -> slaves of bank A from 0 to 15, respectively, provided.
%KWr.m.c.2.n	BOOL	R	n = 0 to `15 -> slaves of bank A from 16 to 31, respectively, provided.
%KWr.m.c.3.n	BOOL	R	n = 0 to 15 -> slaves of bank B from 0 to 15, respectively, provided.
%KWr.m.c.4.n	BOOL	R	n = 0 to 15 -> slaves of bank B from 16 to 31, respectively, provided.
%KWr.m.c.5 to %KWr.m.c.36	INT	R	Words 5 to 36 -> of the slaves of **bank A** from 0 to 31, respectively. Byte 0: bit 0 to 3 = configuration code for input/output (I/O) channels, bit 4 to 7 = identification code (ID), Byte 1: bit 0 to 3 = identification code (ID1), bit 4 to 7 = identification code (ID2).
%KWr.m.c.37 to %KWr.m.c.68	INT	R	Words 5 to 36 -> of the slaves of **bank B** from 0 to 31, respectively. Byte 0: bit 0 to 3 = configuration code for input/output (I/O) channels, bit 4 to 7 = identification code (ID), Byte 1: bit 0 to 3 = identification code (ID1), bit 4 to 7 = identification code (ID2).
%KWr.m.c.69.0	BOOL	R	If this = 1, automatic addressing is active, if this = 0, automatic addressing is deactivated.

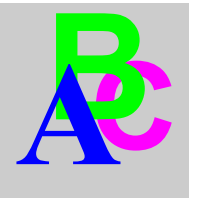
18.4 IODDT T_GEN_MOD applicable to all the modules

Details of the language objects of the IODDT of the type T_GEN_MOD

At a Glance

All the modules of the Premium PLC's have an associated IODDT of the type T_GEN_MOD (See *The IODDT type T_GEN_MOD applicable to all modules , p. 187*).

Index



A

Add a Slave
AS-i Bus, 121
Adding a profile
AS-i Bus, 129
Adding a slave
AS-i V2 Bus, 207
Adding an
AS-i V2 Bus profile, 216
Adjustment
AS-i bus, 140
AS-i V2 bus, 231
Advanced operation
AS-i bus, 152, 153
AS-i V2 bus, 243, 244
Analog Parameters
AS-i V2 Bus, 220
Analog Value Modification
AS-i V2 Bus, 235
AS-i Bus, 105
AS-i bus components, 18
AS-i slave
AS-i bus, 112
AS-i V2 bus, 196
AS-i V2 slave
AS-i V2 bus, 198
Automatic slave addressing
AS-i bus, 131
AS-i V2 bus, 218

C

Characteristics
AS-i bus, 23
AS-i bus power supply units, 37, 55
TSX SAY 100, 70
TSX SAY 1000, 93
characteristics
AS-i V2 bus, 25
Combination of parameters
AS-i V2 Bus, 221
Command objects
AS-i V2 bus, 277
Commands
AS-i bus, 184
Configuration screen
AS-i bus, 119
AS-i V2 bus, 205
Connecting
AS-i bus, 134
AS-i V2 bus, 224
Connection
TSX SAY 100, 64
connection
TSX SAY 1000, 88
Connection of a Device
AS-i Bus, 121
Connections
TSX SUP A02, 46
TSX SUP A05, 48

D

Data exchange off mode
AS-i bus, 153
AS-i V2 bus, 244
Debug
AS-i bus, 134
AS-i V2 bus, 224
Debug Screen
AS-i V2 Bus, 225
Debug screen
AS-i bus, 135
Description
TSX SUP A02, 34
TSX SUP A05, 35
Device Connection
AS-i V2 Bus, 207
Diagnostic
TSX SAY 100, 73
TSX SAY 1000, 95
Diagnostic mode
AS-i bus, 137
Diagnostics
AS-i bus, 137
AS-i V2 Bus, 227
Diagnostics mode
AS-i Bus, 139
AS-i V2 Bus, 227
Display
TSX SAY 100, 76
TSX SAY 1000, 91

E

Environmental characteristics
AS-i bus power supplies,, 57
External faults
AS-i bus, 137
AS-i V2 bus, 227

F

Fallback to 0
AS-i bus, 132
Faulty slave
AS-i bus, 143
AS-i V2 bus, 236
Forcing
AS-i bus, 141
AS-i V2 Bus, 233

G

General
AS-i bus, 16
AS-i bus V2, 16

I

I/O display
TSX SAY 100, 77
Implementation
AS-i bus, 108
Installation
AS-i V2 bus, 192
TSX SAY 100, 63
TSX SAY 1000, 87
Internal errors
AS-i Bus, 137, 227
Introduction
AS-i bus power supply units, 33
IODDT
AS-i bus, 159
AS-i V2bus, 257
Communication, 169, 259
Module, 187

L

Language Objects
Implicit Exchange, 162
Language objects
configuration, 186, 278
Explicit exchange, 163, 171, 261, 274
Explicit exchanges, 177, 269
Implicit exchange, 170, 179, 260, 264, 271
Implicit exchanges, 174
Management of exchanges, 165

M

Maintain
AS-i bus, 132
Master/Slave
AS-i bus, 110
AS-i V2 bus, 194
Methods of fixing
AS-i power supply bus, 44
Miscellaneous objects
AS-i bus, 182
Module selection
AS-i bus, 118
AS-i V2 bus, 204
Mounting
TSX SUP A02, 40
TSX SUP A05, 43

O

Offline mode
AS-i bus, 152
AS-i V2 bus, 243
Operating mode
AS-i bus, 148
AS-i V2 bus, 240
Operating modes
TSX SAY 100, 79
TSX SAY 1000, 97

P

Parameter Modification
AS-i V2 Bus, 219
Parameter objects
AS-i V2 bus, 277
Parameters
AS-i bus, 184
Performances
AS-i bus, 155
AS-i V2 bus, 245
Physical description
TSX SAY 100, 61
TSX SAY 1000, 85
Precautions, 52
Profile
AS-i bus, 110
AS-i V2 bus, 194
Project browser
AS-i bus, 124
AS-i V2 bus, 211
protégé mode
AS-i V2 bus, 242
protégémode
AS-i bus, 150

R

RESET
AS-i bus, 142
AS-i V2 bus, 234

S

Safety
TSX SAY 100, 71
SAFETY_MONITOR
AS-i bus, 248
Bus AS-i V2, 248
Security Device
AS-i V2 Bus, 222
SET
AS-i bus, 142
AS-i V2 bus, 234
Slave Address Modification
AS-i V2 Bus, 238
Slave configuration
AS-i bus, 119
AS-i V2 bus, 205
Slave Insertion
AS-i V2 Bus, 237
Slave number
AS-i Bus, 123
AS-i V2 bus, 209
Slaves
AS-i bus, 139
AS-i V2 Bus, 229
Slaves' status
AS-i V2 Bus, 229
Status of the slaves
AS-i bus, 139

Structure of a standard address AS-i V2
    AS-i V2 bus, 196
Structure of an AS-i slave
    AS-i bus, 112
Structure of an extended address AS-i V2
slave
    AS-i V2 bus, 198
Support board
    TSX SUP A02, 36

T

T_COM_ASI_V1
    AS-i bus, 174, 177
T_COM_ASI2
    AS-i bus, 264
    AS-i V2 bus, 269
T_COM_STS_GEN, 170, 171, 260, 261
T_GEN_MOD, 187
test de cablâge mode
    AS-i bus, 151
TSX SAY 100, 59
    Addressing of I/O's, 114, 200
    AS-i bus, 110
    Display, 67
TSX SAY 100 module
    AS-i bus, 110
TSX SAY 1000, 83
    AS-i V2bus, 194
    installation, 87
TSX SAY 1000 module
    AS-i V2 bus, 194
TSX SUP A02, 31
TSX SUP A05, 31

U

Unforcing
    AS-i bus, 141
    AS-i V2 Bus, 233